Neural modeling of human motor coordination inspired by biological signals aiming for parkinsonian gaits

Andrii Dmytrovych Shachykov

▶ **To cite this version:**

Andrii Dmytrovych Shachykov. Neural modeling of human motor coordination inspired by biological signals aiming for parkinsonian gaits. Computer Science [cs]. Université de Lorraine; Institut polytechnique de Kiev (Ukraine), 2019. English. NNT : 2019LORR0291 . tel-02735309

HAL Id: tel-02735309
https://hal.univ-lorraine.fr/tel-02735309

Submitted on 2 Jun 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4
Code de la Propriété Intellectuelle. articles L 335.2- L 335.10
http://www.cfcopies.com/V2/leg/leg_droi.php
http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm

UNIVERSITÉ DE LORRAINE

École doctorale IAEM Lorraine

Neural modeling of human motor coordination inspired by biological signals aiming for parkinsonian gaits

THÈSE

présentée et soutenue publiquement le 17 décembre 2019

pour l'obtention du

Doctorat de l'Université de Lorraine

(mention informatique)

par

Andrii Shachykov

Composition du jury

Président :	Yannick Aoustin
Rapporteurs :	Tetiana Aksenova, CEA Grenoble Yannick Aoustin, Université de Nantes
Examinateurs :	Patrick Hénaff, co-directeur de thèse, Université de Lorraine Oleksandr Shulyak, co-directeur de thèse, NTUU "Igor Sikorsky Kyiv Polytechnic Institute" Christine Azevedo, Laboratoire d'Informatique et de Robotique de Montpellier Julien Frere, Université de Lorraine Anton Popov, NTUU "Igor Sikorsky Kyiv Polytechnic Institute"

Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503

Mis en page avec la classe thesul.

Remerciements

This work was supported by ERASMUS+ program between National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute" and Université de Lorraine and French embassy of Ukraine.

I am immeasurably grateful to my family, friends, and colleagues from both Ukraine and France for supporting me during this uneasy double diploma program.

Notably, Patrick Hénaff for supervising me in France and helping with both scientific and administrative issues.

Equally, Oleksandr Shulyak for supervising me in Ukraine and guiding me through PhD studies and administrative tasks.

Anton Popov for his help with student exchange program and co-tutelle agreement between France and Ukraine.

Contents

List of Figures

Glossary

BG : basal ganglia
BSN : brainstem nuclei
CLR : cerebellar locomotor region
CPG : central pattern generator
Cx : cerebral cortex
DA : dopamine
DBS : deep brain stimulation
DCM : dynamic causal modeling
DoF : degree of freedom
ECoG : electrocorticography
EEG : electroencephalography
FA : Feet Adjacent event of gait cycle
fMRI : functional magnetic resonance imaging
FoG : freezing of gait
GABA : gamma-Aminobutyric acid
GEN : Go/Explore/NoGo
GPe : globus pallidus external
GPi : globus pallidus internal
HR : Heel Rise event of gait cycle
Hyp : hypothalamus
IC : Initial Contact event of gait cycle
LFP : local field potential
MLR : mesencephalic locomotor region
MTP : metatarsophalangeal
OHR : Opposite Heel Rise event of gait cycle
OIC : Opposite Initial Contact event of gait cycle
OTO : Opposite Toe Off event of gait cycle
OTV : Opposite Tibia Vertical event of gait cycle
PD : Parkinson's disease
PET : positron emission tomography
Pfn : parafascicular nucleus
MPRF : medial pontine reticular formation

RF : reticular formation
SMA : supplementary motor area
SNc : substantia nigra compacta
SNr : substantia nigra reticulata
STN : subthalamic nucleus
Str : striatum
Th : thalamus
TO : Toe Off event of gait cycle
TV : Tibia Vertical event of gait cycle
VLo : ventrolateral thalamus

Introduction

During our normal everyday life, we constantly walk back and forth without realizing all the background work done by peripheral nervous system. The understanding of this automatic control is complicated, because it is integrated into our decision-making systems and is highly flexible to be reliable and provide our ancestors' survival.

The modeling of the human motor control system is a challenging problem, it includes deep parts of the brain as well as insufficiently studied nervous structures of spinal cord. Obtaining signals from deep brain structures is non-trivial and have to rely on existing implanted electrodes as of Deep Brain Stimulation for humans. And on the bottom of that, we have a multi-body mechanical system, actuated by numerous muscles with non-linear characteristics.

The manifestation of motor control as gait is most noticeably affected by neurodegenerative diseases such as Parkinson's. The effect lays in a specific condition when the lower motor system stays intact by the disease and only the descending control signals result in witnessed symptoms of gait deficiency [1].

Due to this factor of neurodegenerative causes in the brain, researches that consider the other components of motor system are sparse. And usually, the considered lower components of motor control are simple, sought only to evaluate general effect of disease on movement.

To simulate the effect of neurodegenerative diseases on human gait in detail, we would need models of three levels of motor control. Fig. 1 depicts these levels and connections between them. The top level (Basal Ganglia) denotes brain structures, the ones that disrupt control signals. This level controls only the second level, Central Pattern Generators. The latter denotes spinal nervous networks that excitate muscles and give feedback about their state to the brain. And the last level, Musculoskeletal limbs, denotes musculoskeletal system. This level sends feedback about limb and muscle state as well as their interaction with the environment to both second and first level of control.

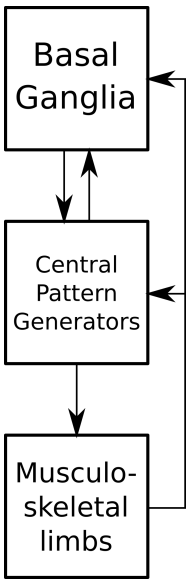


Figure 1: The three levels of gait control.

In this thesis work, a platform for simulating the effects of various abnormal states on gait is proposed. The platform includes bio-inspired models of spinal neuron networks, known as central pattern generators (CPG) [2], musculoskeletal system of human, as well as simulation of inter-body physics and environment, resulting in a neuro-musculoskeletal simulator of human locomotor system.

The platform is constructed to be modular, with interchangeable components of varying complexity. Musculoskeletal system is presented as two models of human lower

body. Simpler GAIT2DE model [3], simulated in MATLAB with 16 muscles in sagittal plane. And a more complex human body model by Rajagopal et al. [4] in OpenSim simulator [5] with 80 muscles of legs. It is a more precise musculoskeletal model based on anatomical measurements and accurate muscle model.

The spinal controller is independent from the musculoskeletal model. On one hand this declares flexibility, on the other hand, the computation time increases, comparing to the implementation as a custom set of differential equations.

In this thesis, we present three architectures of spinal networks based on an original CPG model [6]. It is a bio-inspired mesoscopic CPG architecture adopted from spinal neuron structure studies. Each network architecture has different control target, as well as a supportive reflex controller. We tend to use only biologically available feedback data, i.e. muscle afferent signals. The platform also contains means for gait evaluation and controller tuning.

Our spinal controller is intended to eventually augment or replace the muscle-actuated locomotor system of human. But despite of being one of the final goals of the work, the utilization of actual human muscle system lays in the far future and anyway is unnecessary on current level of technology.

On the other side from surgical implantation, robotic control is quite approachable, but tendon-actuated robots are few [7, 8] due to greater mechanical complexity. Usually, they are presented as pulley-actuated hand-like manipulators [9].

Utilizing computer simulation instead offers two unique features — determinism and computability. First provides the researcher with the same results between trials. The second feature allows to compute and use any value for the means of control and result evaluation. Thus, this platform is developed to reproduce signals during healthy or altered walking gaits.

The main contributions of the work, presented in this thesis are as follows. A short review of models of Parkinson's disease-affected structures in the brain that may be applicable as decision-making module in simulating of motor system of human impaired by various neurodegenerative diseases. Then, we propose three alternative architectures of bio-inspired spinal muscle control networks, each of different control target. These networks acknowledge for decision-making input and produce neuronal excitation directly for each muscle in a closed loop utilizing biologically plausible afferent feedback.

Combined with the description of using two musculoskeletal models of human locomotor system, a simpler one in MATLAB, and a more complex and anatomically correct model in OpenSim simulator, the work formulates a modular platform for simulations of healthy and impaired human gaits. Finally, we demonstrate various gait simulations utilizing our platform.

The thesis work is structured as follows. Chapter 1 describes human gait, it's neuronal control, normal state of gait and abnormalities due to neurodegenerative diseases, focusing on Parkinson's disease and, finally, the modeling of such impairments on bio-inspired neuronal networks.

Chapter 2 tells about central pattern generators, their biology and modeling. This chapter also introduces our original CPG model and its application for a spinal controller. And mentions other controllers that support spinal gait.

Chapter 3 elaborates on muscular and skeletal models of human body and the concrete implementations used in this work. This chapter describes two used musculoskeletal models and required changes for integration with our controller as well as concrete implementation of the simulation platform.

Chapter 4 considers simulated gait analysis, consisting of simulation comparison with parameters of real gait, details of such comparison, and controller optimization based on comparative analysis.

Chapter 5 presents results of simulations of human gait with CPG controller in different architectures in seven simulation setups. The controllers differ in control target, upper controller involvement, and the used musculoskeletal model.

Chapter Conclusion finalizes the work and discusses its results and future work in the field of neuro-musculoskeletal human gait simulation.

Chapter 1

Healthy and deficient gait

The gait is a crucial part of our well-being. Due to its bipedal nature, locomotion control has grown complex, supplemented with balance maintaining. The locomotor system contains many levels and subsystems to control the movements. From the brain cortex commands to the muscular and skeletal architecture. Because of complexity and multitude of compensatory mechanisms of our locomotion, gait impairments of any level result in overall gait abnormality.

Simulation of the whole human locomotor system helps to deepen our understanding of the human body functioning. More common and complex means of simulation profit the analysis of walk gaits and thus the non-invasive diagnosis and treatment.

By simulation of human locomotor system, this thesis aims to define a neural controller able to activate simulated muscles in a closed loop, get feedback from muscular activity and interact with the environment.

But prior developing the neural controller, we must survey the expected results and involved biological structures. This is the goal of this chapter.

1.1 Classification of gaits

The expected result of our work would be the platform able to generate different human gaits, both normal gait and various kinds of impaired gaits. The simulated gaits will be further compared to real ones and analyzed.

The process of simulation provides us with physical parameters of gait, such as joint angles and limb coordinates. Thus we have two options to approach these signals' prior processing. Either, we can transform physical values into well-known gait features and impairment symptoms. Or we can use a table, where gait features and symptoms match physical gait values.

Unfortunately, the second option, the matching table, is not accessible. The closest analogues we have found are just two sources with lists and charts of gait abnormalities with approximate symptoms and gait features. None of them include physical gait values that correspond to given symptoms.

All the more so, the clinical terminology and diagnostics nowadays still lack the clearness and unification of impaired gait description [10, 11].

These two following works show the value and validity of our first approach, namely to simulate gait using a neural controller with enforced neural or musculoskeletal symptoms.

Stanford Medicine

The website of Stanford Medicine[1] provides brief description of eight neurological gait abnormalities. These abnormal gaits are: Hemiplegic, Diplegic, Neuropathic, Myopathic, Choreiform (Hyperkinetic), Ataxic (Cerebellar), Parkinsonian, and Sensory.

The work describes gaits verbally, using terms like "leg flexed" or "dragged in semicircle". Such description, while being relevant in diagnostics, need means to recognize these symptoms in measured or simulated gait.

Work of Azulay et al.

This work in French [12] provides us with wholesome description of human locomotion and sources of impairments. Most relevant, it contains a chart with disordered gait classification. Fig. 1.1 reproduces the said chart with our translation of symptoms and features.

The work of Azulay et al. as well provides only clinical diagnoses of gait disturbances and features, meaning a good amount of work to bring its results to simulated gait analysis.

These two works does not contain physical values of abnormal gaits. Thus, the solution lays in an independent work to construct such a table of physical values tied to gait disorder symptoms. Our work does not conduct such task and leaves it for the future. The current thesis work evaluates resulting gaits by comparing them to the real ones.

1.2 Motor control

The neural control of human gait, or motor control, is initiated in the brain [13, 14]. The exact way of controlling the locomotion is unknown, we can only operate with hypothetical pathways and corresponding theories.

Fig. 1.2 shows hypothetical pathways in the brain and spinal cord. According to this work, the cerebral cortex structures (on top of the figure) initiate movement both directly and indirectly, through mediating basal ganglia (structures in the center of the figure).

While the cortex is responsible for gait initiation, direction, and speed, basal ganglia selects the locomotor behavior and integrates sensory data, obtained from cerebellum. Further, brainstem nuclei (MLR, MPRF) serve as a hub for brain signals to maintain equilibrium and modulate the ongoing locomotor CPG activity. Networks in the spinal cord, CPG, receive input from medial pontine reticular formation (MPRF), which is mediated by mesencephalic locomotor region (MLR).

1.2.1 Basal ganglia

Basal ganglia is a set of seven nuclei in the midbrain. BG is involved in different low-level brain functions, relevant to action selection and reward learning [16]. These nuclei have their own hypothetical relations

[1] https://stanfordmedicine25.stanford.edu/the25/gait.html

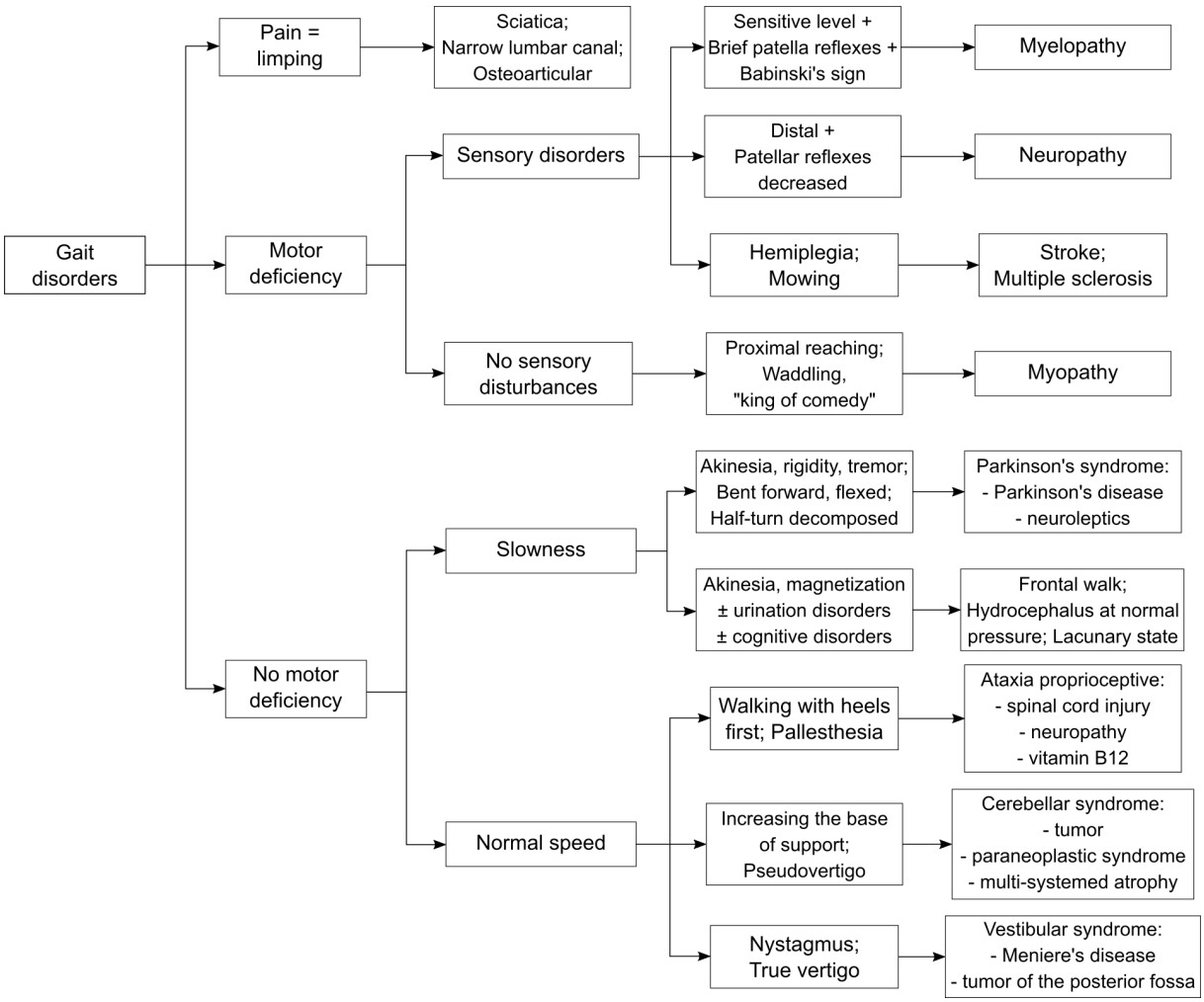


Figure 1.1: Disordered gait classification. Remade from [12] and translated.

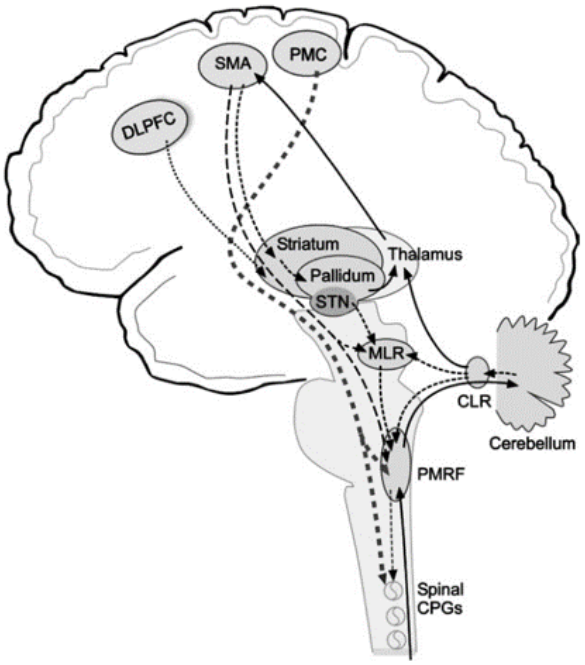


Figure 1.2: Hypothetical motor pathways in the brain, extracted from [15].

and connections. Fig. 1.3 shows a joint basal ganglia connection scheme, that we have compiled from several works [16, 17, 18, 19, 20, 21, 22, 23].

Basal ganglia sends commands to central pattern generators through mesencephalic locomotor region (MLR), reticular formation, and extrapyramidal tracts in brainstem [14].

The basal ganglia consists of the following nuclei. Globus pallidus (GP) is a structure in the brain involved in the regulation of voluntary movement. It consists of two distinguished regions, internal (GPi) and external (GPe) parts [17]. When it comes to regulation of movement, GP has a primarily inhibitory action that balances the excitatory action of the cerebellum. These two systems are designed to work in harmony with each other to allow people to move smoothly, with even, controlled movements.

The substantia nigra (SN) plays an important role in reward and movement. It consists of two parts with very different connections and functions: the pars compacta (SNc) and the pars reticulata (SNr). SNc serves mainly as an input to the basal ganglia circuit, supplying the striatum with dopamine. It plays an indirect role by regulating the more direct role of the striatum, contributing to fine motor control. Parkinson's disease is characterized by the death of dopaminergic neurons in this region. SNr serves mainly as an output, conveying signals from the basal ganglia to numerous other brain structures. The neurons of the pars reticulata are fast-spiking pacemakers, generating action potentials in the absence of synaptic input. Thalamus is generally believed to act as a relay between different subcortical areas and the cerebral cortex.

Basically, researchers distinguish two connectivity pathways in basal ganglia, direct and indirect. Direct pathway is thought to facilitate the initiation and execution of voluntary movement. And indirect pathway prevents unwanted muscle contractions from competing with voluntary movements.

The direct pathway connections run as follows (Fig. 1.3). The cortex stimulates striatum, the input

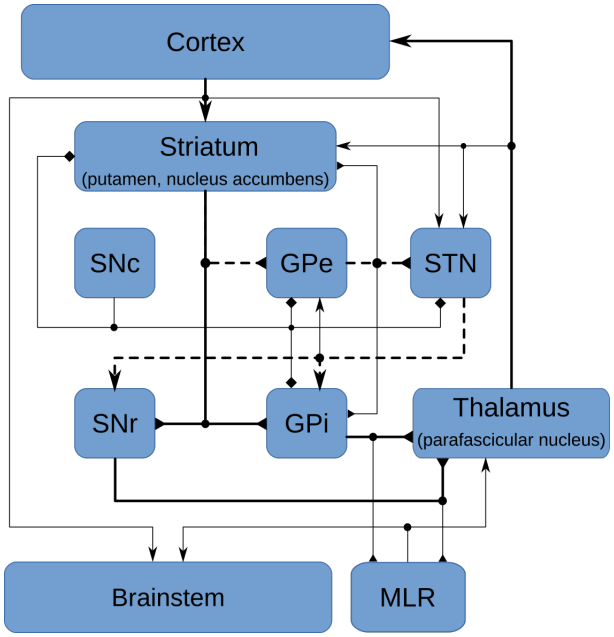


Figure 1.3: Overall scheme of connections in basal ganglia. The direct pathway is denoted by thicker lines and indirect is dashed. Excitatory glutamatergic connections as arrows, inhibitory GABAergic as triangles, modulatory dopaminergic as diamonds. Dots are connection branching. See abbreviations in Glossary on page 1.

hub to BG. Striatum inhibits SNr-GPi complex, which, in turn, inhibits thalamus. Thalamus, in feedback, stimulates cortex, which also stimulates lower brain structures that eventually reach muscles.

The indirect pathway is as follows. After cortex, again, stimulates striatum, the latter inhibits GPe. GPe inhibits STN that stimulates SNr-GPi complex that (as we know) inhibits thalamus. Thalamus, again, stimulates cortex. And so on to muscles.

Thus, to simulate the behavior of brain-related structures to serve as a decision-making of a gait controller, we could content ourselves with only functional simulation of aforementioned structures and pathways. But this relates just to simulation of healthy gait. Simulate brain-caused gait impairments, we must account for the causes of the considered brain dysfunctions.

1.3 Parkinson's disease

In this thesis work, the Parkinson's disease has been considered for simulation of neuronal effects on gait. The reason behind this decision is that this disease noticeably affects gait, through both tremor and the symptom known as Freezing of Gait (FoG). Also, Parkinson's disease has a noticeable cause in the basal ganglia region of the brain. Basal ganglia was described in the previous section as the main region of locomotion decision-making, being crucial in gait simulations.

Parkinson's disease (PD) is a progressive neuro-degenerative illness with symptoms like tremor, muscular rigidity, and slow, imprecise movement mainly affecting elderly individuals over 60 years old [24]. PD affects control of movements, especially upper and lower limbs, causes tremor and rigidity [25]. In

2015, PD affected 6.2 million people and resulted in about 117,400 deaths globally [26]. PD incidence generally increases with age: from 3 per 100,000 person-years at age 40 to 258 at age 80+ [24]. It is the second most common neuro-degenerative disease of adult onset [27].

PD's symptoms may be divided onto two groups, behavioral problems and movement ones. Behavioral problems include dementia and depression occurring in the advanced stages of the disease, sensory, sleep, and emotional problems, and general thinking and behavioral problems. The symptoms group, relevant to this work, consists of shaking, rigidity, slowness of movement and difficulty with walking with noticeable FoG symptom [15, 28].

Freezing of Gait is a specific symptom, observable in patients with Parkinson's disease. It is characterized by short episodes of either inability to make a step or very short steps during initiation of gait or turning [15]. This symptom is relatively easy to study and even counteract by applying various cues for the patients [29, 30, 31, 32, 33].

1.3.1 Anatomical pathology of Parkinson's disease

The basal ganglia, innervated by the dopaminergic system, is the most seriously affected brain area in PD. The recognized cause of PD is the death of dopaminergic neurons in the substantia nigra pars compacta (SNc) region of basal ganglia in the midbrain [25]. Secretion of neuro-mediator dopamine (DA) in this region leads to decreased inhibition of motor system, releasing it for activation. The outcome of DA loss is increased effort to move [16].

There are five major pathways in the brain connecting other brain areas with the basal ganglia. These are known as the motor, oculo-motor, associative, limbic, and orbitofrontal circuits, with names indicating the main projection area of each circuit. All of them are affected by PD, and their disruption explains many of the symptoms of the disease since these circuits are involved in a wide variety of functions including movement, attention and learning. The pathological characteristic of PD of cell death in the SNc that plays an important role in reward and movement, affecting up to 70% of the cells by the time when patient's death occurs [34].

Normally, the basal ganglia exert a constant inhibitory influence on a wide range of motor systems, preventing them from becoming active at inappropriate times. When a decision is made to perform a particular action, inhibition is reduced for the required motor system, thereby releasing it for activation. Dopamine (DA) acts to facilitate this release of inhibition, so high levels of DA tend to promote motor activity, while low levels of DA, such as occur in PD, demand greater exertion of effort for any given movement.

Thus, the net effect of DA depletion produces hypokinesia (decreased bodily movement), an overall reduction in motor output. Drugs that are used to treat PD, conversely, may produce excessive DA activity, allowing motor systems to be activated at inappropriate times and thereby producing dyskinesia (involuntary muscle movements).

The motor symptoms of the PD result from the death of DA neurons in the SNc. In the absence of dopamine, D1-receptors in the basal ganglia don't stimulate the GABAergic neurons, which favor the direct pathway, and thus not increasing movement. This sets off the indirect pathway that results in inhibition of upper motor neurons and less movement.

In the presence of DA, D2-receptors in the basal ganglia inhibit these GABAergic neurons, which

reduces the indirect pathways inhibitory effect. The antagonistic functions of the direct and indirect pathways are modulated by the SNc, which produces dopamine.

In the presence of DA, regardless of its source, D1-receptors in the BG stimulate the GABAergic neurons, favoring the direct pathway, and thus increasing movement. The GABAergic neurons of the indirect pathway are stimulated by excitatory neurotransmitters acetylcholine and glutamate. This sets off the indirect pathway that ultimately results in inhibition of upper motor neurons, and less movement. In the presence of dopamine, D2-receptors in the basal ganglia inhibit these GABAergic neurons, which reduces the indirect pathway's inhibitory effect.

DA, therefore, increases the excitatory effect of the direct pathway (causing movement) and reduces the inhibitory effect of the indirect pathway (preventing full inhibition of movement).

Through these mechanisms, the body is able to maintain balance between excitation and inhibition of motion. Lack of balance in this delicate system leads to pathology such as Parkinson's disease. During PD, the direct pathway is less able to function (so no movement is initiated) and the indirect pathway is in overdrive (causing too much inhibition of movement).

1.3.2 Models of neuron circuitry with implied Parkinson's disease

Considering the functioning of basal ganglia affected by Parkinson's disease, the researchers had made numerous models that simulate PD either only in brain structures or applied to a certain task. But most of the models are limited to BG and relations between its nuclei. These models result in firing rates of neurons or other intrinsic parameters of system as their aim is to identify the source of abnormal oscillations in BG, related to PD. Human movements governed by lower structures that receive signals from BG are as well perturbed by PD and this impacts the daily life of affected people. Therefore, despite interesting ways of modeling BG and PD, these works are not sufficiently connected to consequences of PD on the human movements and especially on the walk.

Nevertheless, the models of subcortical structures (sometimes including motor cortex) vary in complexity, fitting their research purpose. The models that look for the origin of parkinsonian tremor [23, 22] result as firing rates of neurons, which, for the control purpose, need to be transformed into commands for lower structures. Other models are more applicable and simulate handwriting [19], object lifting [35], and walking [36].

In the considered models, PD simulation is applied through special dopamine level parameter or fitting the parameters to experimental data of parkinsonian patients. In the following, we describe these six representative models of brain structures including basal ganglia affected by PD.

1.3.2.1 PD tremor model

In the work [23], the authors have searched a source of PD tremor in Basal Ganglia-Thalamo-Cortical circuit, simplified to conductance-based model of 3 biological neurons. Fig. 1.4 shows the said network. There are one GPe and one STN neurons and a feedback neuron, represented by a blank box.

PD was simulated by using such synaptic strengths that result in tremor-like activity of the model. The model result in Hodgkin-Huxley neuron activity that has to be non-trivially further transformed in control commands for motor network.

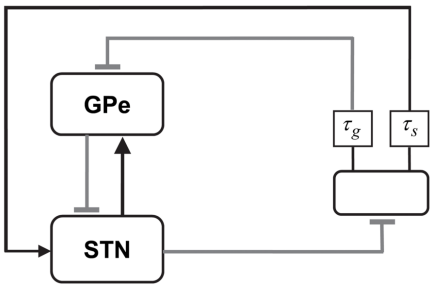


Figure 1.4: The model's circuit, extracted from [23]. Arrows indicate excitatory synapses and bars indicate inhibitory synapses. Squares indicate the delay units with the delays τ_s and τ_g.

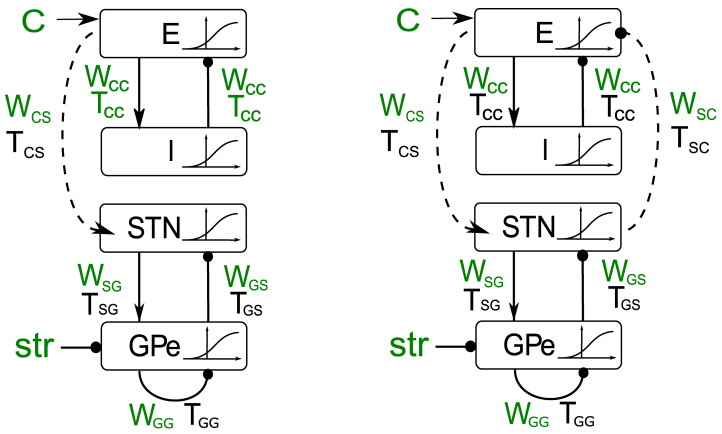


Figure 1.5: Computational models of cortico-basal-ganglia-thalamic circuit, extracted from [22]. Left: resonance model of the circuit. Right: feedback model, similar to the resonance one, except the feedback loop.

1.3.2.2 PD-related BG oscillations model

In the work [22], the authors have evaluated several computational models of BG with experimental data presented in the paper [37]. It states that for generation of abnormal oscillations only cortex-STN-GPe connections are essential. As a result, they identified two models (Fig. 1.5): a resonance model and a feedback model that both could maintain abnormal beta oscillations. As of results, both models are similarly close to experimental data.

The most important thing is that the MATLAB source code for models is available `http://modeldb.yale.edu/184491`, which simplifies the utilization of this model as a controller for abnormal gait.

PD was simulated with increased synaptic weights values, obtained from parkinsonian monkeys' models. As in previous work, resulting neuron firing rated of this model have to be transformed into motor commands.

1.3.2.3 Data-based PD-related oscillations model

In the work [38], the authors have modeled data from patients undergoing surgery for deep brain stimulation (DBS) of more than one target (scalp EEG and local field potentials (LFP) from DBS electrodes

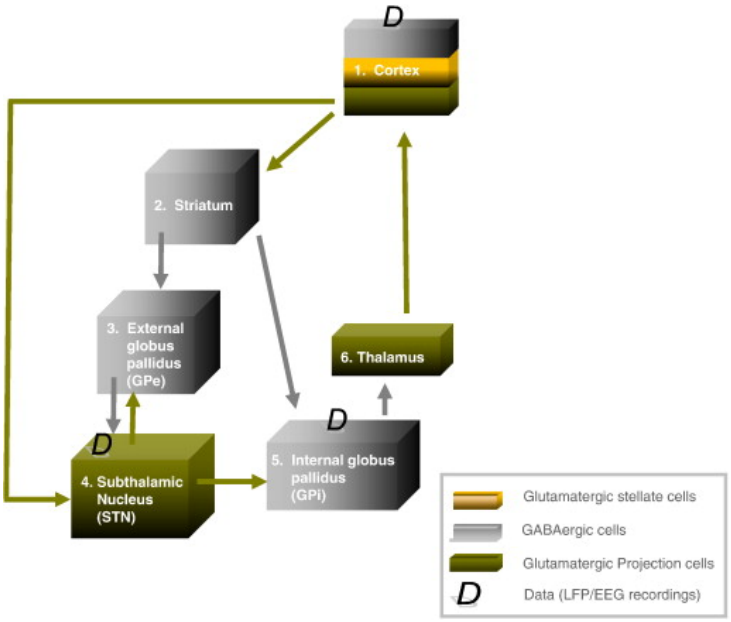


Figure 1.6: The dynamic causal model, extracted from [38]. It shows six modeled nuclei, only three of which had recorded signals, denoted by *D*.

in STN and GPi). They have found that connections to and from the subthalamic nucleus (Fig. 1.6) were strengthened and promoted beta synchrony in the untreated PD, compared to the treated Parkinsonian state. They used dynamic causal modeling, a method to interpret functional neuro-imaging, to replicate the effects of lesion of STN. Using neural mass models that embody ensemble firing output and membrane potential inputs, their model was making valid predictions regarding the consequence of lesions of the subthalamic nucleus and its connections.

PD was simulated by fitting synaptic connection weights to experimental data. Similarly to previous works, this model's resulting neuron firings have to be transformed into motor commands.

1.3.2.4 Handwriting model

Parkinsonism reduces coordination of fingers, wrist, and arm in fine motor control. In [19, 20], authors investigate movement coordination through performing handwriting-like tasks. In PD subjects, tasks involving coordination of fingers and wrist showed larger normalized jerk (quick, sharp movements) than strokes performed using either the wrist or the fingers alone. Same tasks were then simulated using a biologically inspired neural network model.

Fig. 1.7 extracted from the mentioned work, shows the modeled network. It produces relative target pen positions in three axes from the model of part of cortex, basal ganglia, and motor circuits.

PD simulation is performed as DA depletion through a parameter of Hodgkin-Huxley neurons. The work contains a full motor control loop, upper parts of which are usable as a muscle controller.

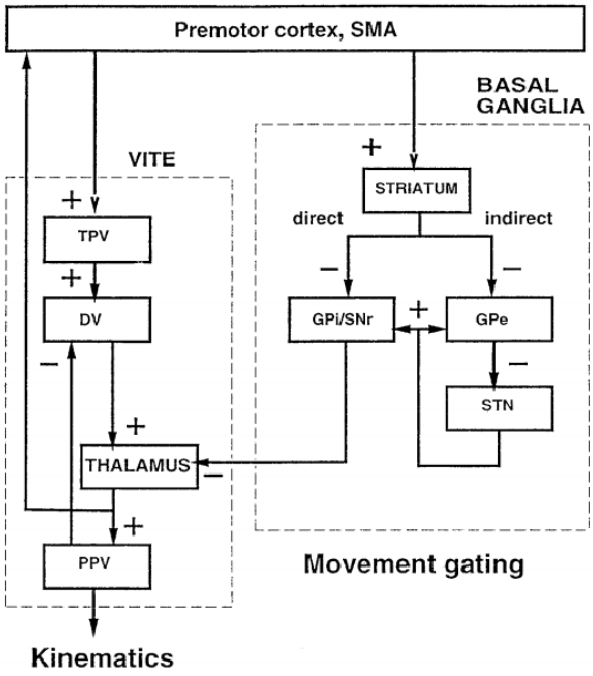


Figure 1.7: Handwriting control scheme, extracted from [20]. DV denotes difference vector of pen position between target position vector TPV and present position vector PPV. VITE is a central movement generator.

1.3.2.5 Precision grip model

In [35], the authors have modeled precision grip of parkinsonian and control subjects. Precision grip is the ability to grip objects between the forefinger and thumb. Their approach was to treat the problem of modeling grip force generation as a problem of action selection GEN (Go/Explore/NoGo). Fig. 1.8 shows a block diagram of their controller.

It is a closed-loop control system with two controllers (grip force, F_G, and lift force, F_L). The proposed model is able to account for the precision grip results from normal and PD patients accurately from two works, where ON and OFF medication and control subjects were lifting 300 g blocks, some covered in silk or sandpaper.

PD simulation is made by attenuating DA signal, which is defined as a function that selects a reference

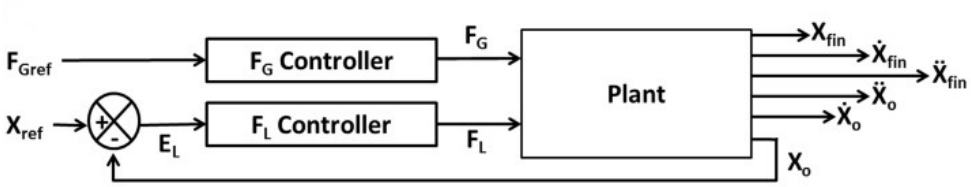


Figure 1.8: The block diagram of precision grip controller, extracted from [35]. F_G is grip force, F_L is lift force. X_o and X_{fin} denote position of a lifted object and a finger, respectively. The plant box denotes a model of gripping and lifting a box with two fingers.

grip force F_{Gref} as an input to the controller. This model is conceptually usable as is as a gait motor controller.

1.3.2.6 PD walk estimation model

In article [36], authors present a computational model of altered gait velocity patterns in PD based on 2 investigations of FoG, as patients at ON and OFF dopaminergic medication walked through a doorway with variable width.

Computational model of BG is based on Reinforcement Learning ("Critic" module), GEN decisions, coupled with a CPG model that mimics spinal rhythm (Fig. 1.9). The Critic module computes the value for the view vector. As a model of the CPG network, the authors used network of coupled non-linear oscillators, modeled using adaptive Hopf oscillators (see Section 2.1.3). The number of Hopf oscillators used to train the hip (ω_h) and knee angles (ω_{k1} and ω_{k2}) are 2 and 3 respectively. Phase difference within-CPGs is maintained by ψ_{local} while across-CPGs is maintained by ψ_{global}. α_s modulate the intrinsic CPG rhythm to output the learned joint angles.

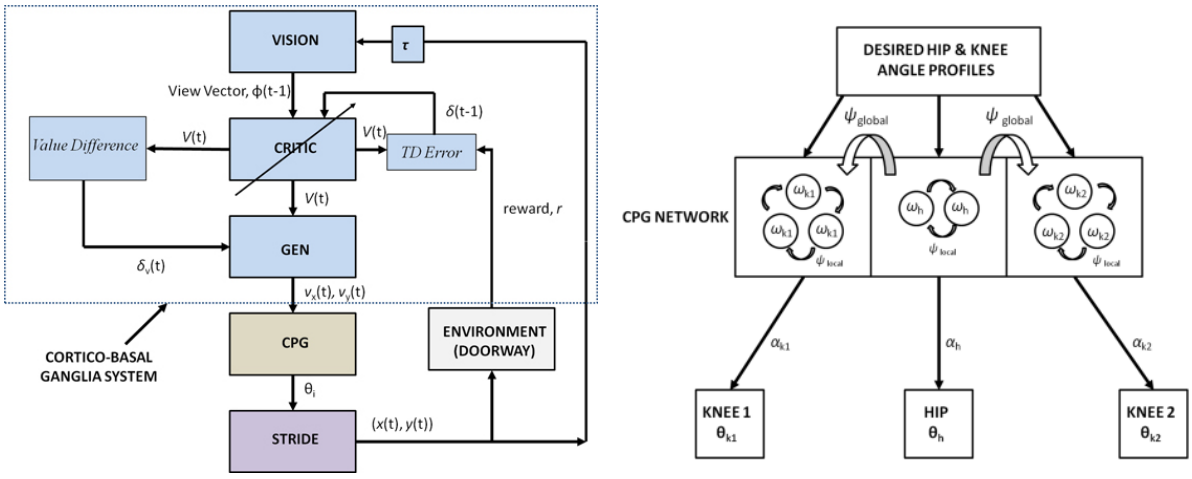


Figure 1.9: Left: Block diagram detailing the Cx-BG system and the CPG module. The diagonal arrow on the Critic represents the module training. Right: CPG network with the desired hip (h) and knee (k1 and k2) angles (θ). Extracted from [36].

Quite similarly to previous work, PD here is simulated through the depletion of DA, δ_V. This model is well suited for integration of neuro-musculoskeletal model of human gait as a decision-making module of motor control.

The presented works model the upper brain structures, affected by Parkinson's disease, but they doesn't consider in-detail modeling of lower neuronal structures, such as central pattern generators, muscles, and skeleton.

Despite the different research purpose, these models are the examples of upper structures controller, able to initiate gait and choose its mode and brain-related abnormal conditions.

1.4 Conclusion

This chapter describes the final goal of this thesis work and details on the origin of motor control and the effects of its impairments on gait. The models of upper motor control consider brain structures such as basal ganglia and may in future serve as decision-making controller for the impaired gait as well as behavior control of healthy gait. Despite not directly damaging locomotor system, PD affects control signals leading to gait problems. To simulate the consequences of those problems we need to simulate central pattern generators and the rest of the neuro-musculoskeletal system of a human.

The central pattern generators, as low-level controllers of gait, are the subject of the next chapter, where we describe them and related modeling.

Chapter 2

Central pattern generators

More than a hundred years of investigations since the famous work of Brown [2] has led to the conclusion that rhythmic activities are largely controlled by network of inter- and motoneurons called CPG (central pattern generator). CPGs for breathing, chewing, and swallowing are located in the brainstem [39] and locomotion CPG are located in the spinal cord [40, 41, 42, 43].

Main feature of CPG is their ability to autonomously generate rhythmic activity, even after being disconnected from upper controller and sensory afferents. This distinctive behavior is repeatedly emphasized in literature [44, 45], though works studying oscillations emerging from positive feedback from external signals exist [46, 47, 48].

Another competing theory is muscle synergy hypothesis [49, 50, 51, 52]. It proposes that complex control signals are built from combination of simpler patterns with their own delay and amplitude. These patterns are generated by specialized interneuron networks in spinal cord. This concept is close to CPG, except synergistic networks don't oscillate and provide their output pattern by request from upper control center.

In this thesis, CPGs are defined as spinal neuron circuits that autonomously produce rhythms that are passed throughout motor neurons to muscles. The control signals from the brain initiate locomotion and change its type. These descending signals are still optional and affect the shape of generated patterns and synchronization, provide equilibrium, choose type of produced gait [53]. Sensory feedback is also optional, but it helps to maintain rhythm and adapt to environment in a closed control loop. Sensory feedback also contributes to inhibition of CPG output, resetting its phase, and other conditional events [54].

2.0.1 Spinal cord and descending control

There isn't much known about connection between the brain and CPG, mostly general connection paths and involved structures. CPG obtain their high-level input through extrapyramidal tracts in brainstem [14]. In this work, we refer to this high-level controller as MLR (mesencephalic locomotor region). This brain region is thought to initiate locomotion in the chain of decision-making regarding autonomous movement [43, 55].

Fig. 2.1 shows general scheme of motor control. CPGs receive input from reticular formation in Brain-

stem, which is mediated by MLR. The latter has projections from basal ganglia and hypothalamus (part of basal ganglia block). Basal ganglia is believed to select the locomotor behavior, while hypothalamus enables it. Brainstem nuclei maintain equilibrium and modulate the ongoing locomotor CPG activity through extrapyramidal tracts in spinal cord.

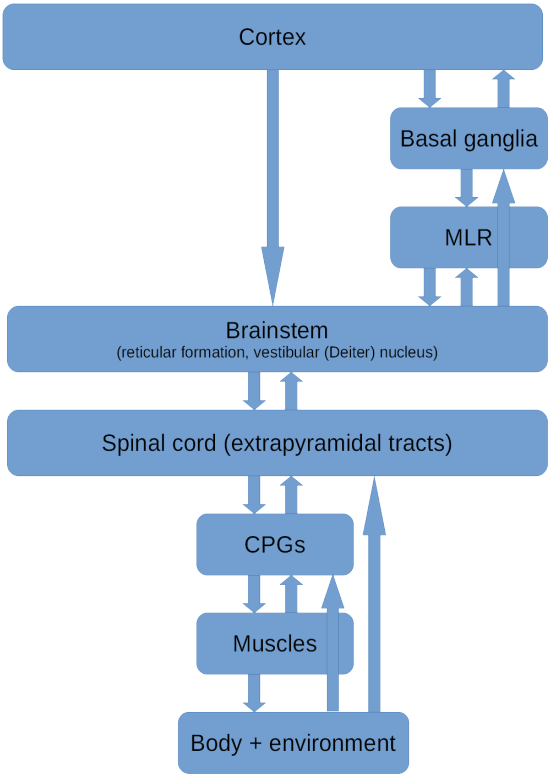


Figure 2.1: Global scheme of movement control. Based on [14].

2.0.2 Feedback signals of motor control

As for the sensory feedback from muscles and environment, we tend to use biologically plausible signals to control human gait on spinal level. Only one of used receptor signals originates in brain. It is vestibular signal from inner ear that is integrated in sensory pathways through brainstem [56] (Fig. 2.1).

The other signals are muscle and cutaneous proprioceptive sensors. They measure inner state of the organism. Simulated in this work sensors reacting to state of the muscles are afferents of type *Ia*, *Ib*, and *II* [14].

Ia-type afferents are muscle spindle stretch receptors that react to muscle length changes and are fast adapting thus providing muscle stretch velocity signal to CPGs and brain. Each *Ia* sensory neuron react to positive contraction velocity of its muscle. Which means that *Ia* sensor indirectly measures opposite muscle's activation velocity. *Ia* sensor excite its own motoneuron to limit joint's angular velocity.

Ib-type afferents are Golgi tendon organs that sense the force generated by muscle and sent it to CPGs and brain.

II-type afferents measure muscle's length and are non-adapting providing CPGs and brain with in-

stantaneous muscle length.

All presented sensors besides of providing feedback to motor controllers, participate in unconditional motor reflexes as well [57]. The problem of separation of these motor actions between CPGs and independent nervous loops is open and is adverted in this thesis.

2.1 Models of central pattern generators

Despite the origination of CPG theory in biology from locomotion experiments, we cannot easily model the network of interneurons to control the locomotion. The biological model of CPG is not established yet and is under active investigation [42, 58, 44, 59]. The researchers have developed a number of models of different levels to mimic CPG's specific features.

2.1.1 Microscopic models

The microscopic models simulate CPG behavior on cellular level. They are being used for investigation of functions and regimes of specific groups of neurons in different segments of spinal cord [14, 39].

These models utilize different variations of Hodgkin–Huxley neuron models for oscillating circuits [60, 61], sometimes deriving special oscillating neurons, like Morris–Lecar model [62]. Hodgkin–Huxley neuron models a cellular membrane as a capacitor, whose changes in voltage are proportional to the sum of currents, flowing through the membrane.

For example, Eq. 2.1, extracted from [61], models four currents of various sources, through ion channels of cell membrane or external currents.

$$C_m\dot{V} = I_{leak} + I_{syn} + I_{NaP} + I_{app} \tag{2.1}$$

where C_m is constant capacitance of cellular membrane, $\dot{V}$ is derivative of membrane voltage, I_{leak} natural leaking current of the membrane, I_{syn} is synaptic channel current, I_{NaP} is current through sodium ion channel, I_{app} is externally applied current.

The number of ion channels may be larger in other models. In the model used by Rybak et al. as Eq. 2.2 [41], there are three ion currents (regular and persistent sodium currents, I_{Na} and I_{NaP} respectively, and potassium current I_K), synaptic excitatory and inhibitory currents, and inverted overall current direction in addition to regular leaky current I_L.

$$C \times \frac{dV}{dt} = -I_{Na} - I_{NaP} - I_K - I_L - I_{SynE} - I_{SynI} \tag{2.2}$$

Microscopic models of CPG are utilized for modeling neural networks resembling natural CPG [41, 63, 64] and further to investigate the neurophysiology of biological CPG [14, 65, 39].

Although our aim is also biological, to control muscles, we sought a more functional approach. Our CPG don't have to be structurally biological, only bio-inspired on a mesoscopic level.

2.1.2 Mesoscopic models

Mesoscopic models simulate the functional behavior of interneuron networks of CPG. They are composed of one or more neurons of unique architecture called Rhythm Generators (RG) and some peripheral or

interneurons around RG. A distinctive feature of mesoscopic models is an interconnected network of two or more RG that produce oscillations.

2.1.2.1 Matsuoka CPG

The popular model of Matsuoka [66, 67, 68, 69, 70, 71] is very popular in robotics. It uses two differential equations (Eq. 2.3) to simulate the firing rate of a neuron. The oscillating unit is composed of several mutually-inhibited neurons with different connection arrangement producing various patterns.

$$\begin{cases} \dot{x}_i + x_i = -\sum_{j=1}^{n} a_{ij} y_i + s_i - b x'_i \\ T\dot{x}'_i + x'_i = y_i \\ y_i = \max(0, x_i) \\ i = 1, \ldots, n \end{cases} \tag{2.3}$$

where y_i represents output of i^{th} neuron of n interconnected neurons, a_{ij} is the strength of the inhibitory connection, s_i is input from the outside of the network, T and b are the parameters that specify the time course of the adaptation.

The model is indeed capable of adapting to the frequency of external input, though its synchronization capability is lower than those of the other CPG models [72].

2.1.2.2 Rowat and Selverston neuron

The model of Rowat and Selverston [40] utilizes generalized Van der Pol oscillator [73] (Eq. 2.4) for the same purpose, to compose an oscillation unit with two RG neurons. It has self-rhythmic generation ability; oscillation depends on two membrane conductivity values for fast and slow current. Depending on these and others cell parameters, neurons can generate different patterns: quiescence, almost an oscillator, endogenous oscillator, plateau, depolarization, and hyper-polarization [53, 6, 74].

$$\begin{cases} \tau_m \dot{V} + V - A_f \tanh\left(\frac{\sigma_f}{A_f} V\right) + q = 0 \\ \tau_s \dot{q} = -q + \sigma_s V \end{cases} \tag{2.4}$$

where V is cell membrane potential, τ_m is membrane time constant, q is a slow current with time constant τ_s and gain σ_s, σ_f and A_f are the parameters of instantaneous current.

This model can accommodate the Hebbian frequency-learning rule [75] to synchronize with an external input, similarly to Matsuoka model. But the dynamic range of synchronization of Rowat and Selverston CPG is higher than that of Matsuoka's [72].

2.1.2.3 Phase oscillator

Ijspeert et al. demonstrated CPG as a system of coupled nonlinear oscillators [76]. Here, the oscillations between bursts of motoneuron activity and periods of rest are modeled by means of a phase oscillator with controlled amplitude (Eq. 2.5).

$$\begin{cases} \dot{\theta}_i = 2\pi\nu_i + \sum_j r_j w_{ij} \sin(\theta_j - \theta_i - \phi_{ij}) \\ \ddot{r}_i = a_i \left(\frac{a_i}{4}(R_i - r_i) - \dot{r}_i \right) \\ x_i = r_i(1 + \cos(\theta_i)) \end{cases} \tag{2.5}$$

where θ_i and r_i are phase and amplitude of i^{th} oscillator, ν_i and R_i are intrinsic frequency and amplitude, a_i is a positive constant, w_{ij} and ϕ_{ij} are weights and phase biases of oscillators' couplings, x_i is a burst signal of the central controller.

2.1.3 Macroscopic models

Going further from biological structure, macroscopic models of CPG offer a whole single unit producing desired patterns.

2.1.3.1 Hopf oscillator

Hopf oscillator was applied by Righetti & Ijspeert [75] as a CPG in controlling a biped robot. It doesn't have "two-center" composition, but capable of frequency learning.

$$\begin{cases} \dot{x} = \gamma(\mu - r^2)x - \omega y + \varepsilon F(t) \\ \dot{y} = \gamma(\mu - r^2)y + \omega x \\ \dot{\omega} = -\varepsilon F(t)\frac{y}{r} \\ r = \sqrt{x^2 + y^2} \end{cases} \tag{2.6}$$

where $F(t)$ is a periodic input with coupling constant ε, y is output, μ controls amplitude of the oscillations, γ controls the speed of recovery after perturbation, ω controls the frequency of the oscillations.

Hopf oscillator, according to Jouaiti et al. [72], is simpler and easier to control than Rowat and Selverston CPG, it has smaller dynamic range and doesn't support integration of additional learning rules, e.g. amplitude learning.

2.1.3.2 State machine CPG

Jo et al. [77] assume CPG as a five-state machine with a strict and fixed sequence of transition that controls five "epochs" of gait cycle: Loading, Regulation, Thrust, Retraction, Forward. Each epoch has its primary function, such as maintaining balance or moving a limb. The functions are mechanistic and straightforward, such as "prevent the upper body from falling backward", "raise leg and trigger forward swing around hip", etc.

Their spinal pulse generator synergistically excite muscles, meaning that each CPG controls several predefined muscles.

In this thesis work, we use a modified version of Rowat and Selverston CPG, with three integrated learning rules, although they are not fully utilized due to several reasons described further.

2.2 Application of CPG models

The models described in Section 2.1 are applied in various fields. Despite the origination of CPG in the analysis of purpose and structure of different groups of neurons in the spinal cord segments, models of artificial CPG are also widely used in bio-inspired control of robots.

The CPG concept is used in biomechanical engineering for synthesis of control units from neural networks resembling CPG to investigate locomotor neurons [64, 41, 58, 59] or animal rhythmic movements [40, 60, 76]. This field generally approach micro- or mesoscopic CPG models as biological structure of interconnected network of simple cells.

Robotic control is a very demonstrative target for CPG controllers as it offers natural kinematic feedback of a robot and profits from robust and flexible bio-inspired control. Although, CPG control is not actively used in commercial robots, aside research.

Specific to movement simulation, closed-loop oscillatory CPG controllers are used to construct bio-inspired controllers for robotic leg [78, 61] or arm [79, 72].

In research of bipedal walking, synthesis of control units that have the same behavior patterns of biological CPGs is extensively used in both commercial and self-made biped robots [75, 80, 81, 82, 74]. An interesting work of Van der Noot et al. [83] uses virtual muscles to calculate robot's joint torques. More on CPG-controlled robots in the work of Yu et al. [84].

There are few works on applying CPG controllers to simulate gait with models of human lower body. Most works consider bipedal walking of three joints of two legs, driven by torque [70, 6, 71]. This approach is similar to robotic control that utilizes motors to rotate joints with a single signal of current. In case of half-center CPG model, such controller deducts flexor and extensor CPG signals to obtain a desired articular value (position, velocity, or torque) applied to the PID controller of joint.

The application of CPG in anatomical gait control is different from torque-actuated gait in robotics. Biological joints are actuated by muscles, thus profiting from half-center model. Half-center models provide two anti-phase outputs from a single oscillation unit, which naturally apply to antagonistic flexor/extensor muscular control.

All the works on CPG gait simulation agree with the necessity to close the control loop with feedback from the biped and environment. But these works satisfy at using kinematic values of joints as feedback for controller, some of which have limited use in our body.

Finally, the narrowest set of works approach to simulate biological values for feedback from muscles [67, 68, 69, 85, 77]. Noticeably, the work of Ogihara et al. [68] is considered the closest to our work, although it has few following differences.

Indeed, their work utilizes one CPG of Matsuoka model with two muscle feedback afferents, lacking type *II* afferent. As of exteroceptors, the model of Ogihara et al. does not simulate balance receptors, yet remarkably achieves dynamic equilibrium during gait.

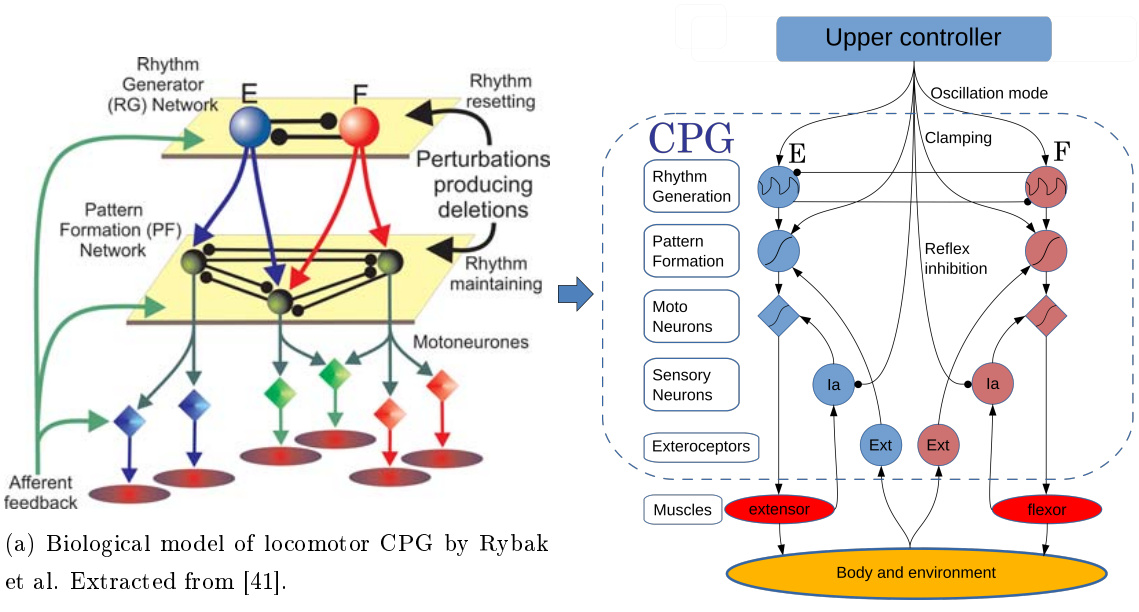
The advantages of our work include general-purpose anatomical model of human body with more accurate muscular geometry and all three types of muscle afferents for feedback. As well as bio-inspired mesoscopic controller, based on the works of Rybak et al. [41, 58] on investigation of human and animal CPG.

2.3 Our model of central pattern generator

Our gait controller utilizes mesoscopic CPG model. The mesoscopic neuron models provide more complex behavior than microscopic, while are easier to tune. On the other hand, they are still close to biology, apart from macroscopic models. Previously, our CPG model has been used to generate patterns for humanoid robot locomotion [53, 82, 74] and more recently for better understanding of robot-human handshake interaction [72, 86]. The model is supported by two neurophysiological studies [41, 87] and combines their propositions in multi-layered multi-pattern CPG model. An important feature of using a CPG as a controller, is its ability to produce repeated patterns without descending signals from brain. This is confirmed in our simulations showing our model of CPG is able to produce stable gait during whole simulation without using input from upper structures.

2.3.1 Overall CPG architecture

This architecture is based on the work of Rybak et al. [41] for a two-level CPG that separates the timing and activation of the locomotion cycle. Fig. 2.2a shows the general CPG scheme with two layers plus motoneurons and afferents with half-center architecture. Oscillations are generated at the top rhythm generator (RG) layer and then passed to pattern formation (PF) layer, which innervates motoneurons (MN). Our CPG model has similar layers (Fig. 2.2b). Additionally, the model explicitly includes upper controller and feedback sensory neurons (SN) that shape the activity of the CPG neurons. These neurons are of two types: proprioceptors of muscle feedback (type-*Ia*, *Ib*, and *II*) and exteroceptors of environment (body angle, foot/ground force).



(a) Biological model of locomotor CPG by Rybak et al. Extracted from [41].

(b) Our CPG model with five layers and upper control.

Figure 2.2: Proposed mesoscopic CPG model, based on microscopic architecture.

The model's modes and activity are controlled from upper brain controller (e.g. MLR) that varies the frequency of generated patterns, phase deletion, and clamping of controlling signal. Descending command signal tonically activates the networks by increasing excitability at both RG and PF levels. On Fig. 2.2b, lines ending with arrow and circle stand for excitatory and inhibitory connections respectively. E and F stand for extension and flexion half-centers that control corresponding muscle groups.

The model synchronizes the movement of both legs in the same way as half-centers, with inhibitory connection, but only between flexion RG neurons, as suggested by biological evidence [88, 54, 58]. This absence of inhibition between extensor half-centers is reasoned by existence of simultaneous stance position of both legs [88].

Fig. 2.3b demonstrates two coupled CPG of legs with connected flexion RG and full set of afferents with hypothetical connections, inspired by biological network in [58] (Fig. 2.3a). This figure shows the connections of exteroceptors as well, with single body angle neuron (FS) and individual ground reaction neurons (GS). The latter may be expanded to several sensors measuring force on different parts of the foot.

The composition and connections of afferent sensors is undefined in general, left alone the best options for this model. Direct innervation of motoneurons from muscle spindles and Golgi tendon organ exist, they may even create required entrainment for gait [54, 89]. Among the tested compositions, we have been using two variants: connecting all afferents inside CPG to motoneurons or using a separate module with reflexes.

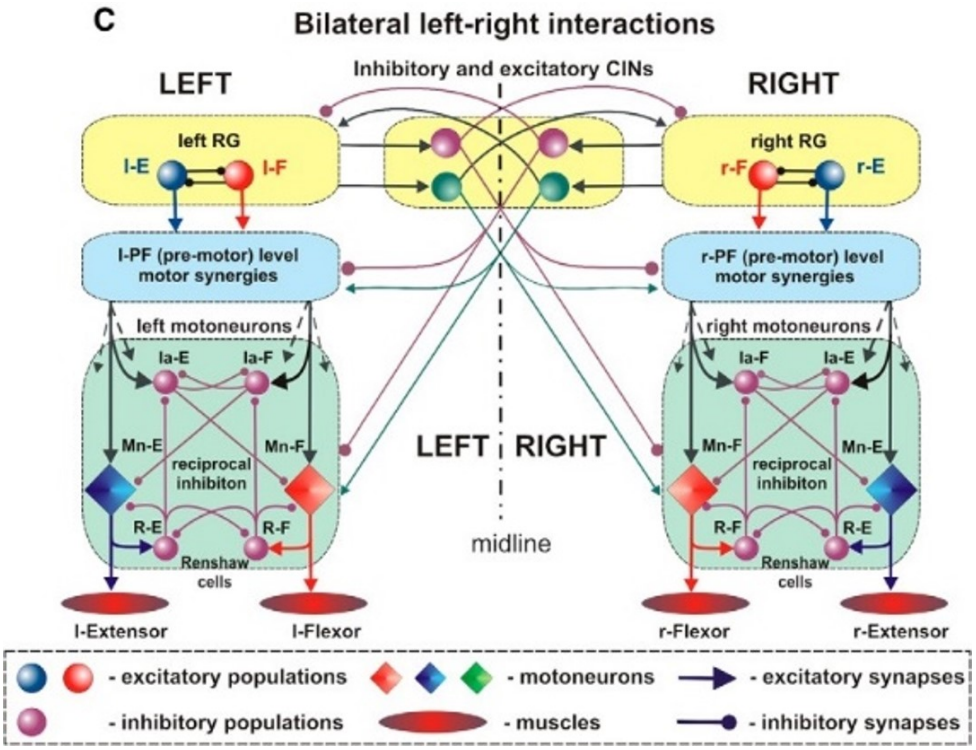
The final CPG arrangement contains only sensors of type-*Ia* as they are thought to participate in autonomous gait by limiting joint angles [54]. The rest of the muscle sensors are implemented in reflex controller (see Section 2.5), where the latter is discussed in application to balance control.

The variants to connect afferents to other layers (RG and PF) are not effected due to differentiation of layer's functions for clearer control synthesis. RG layer is defined to create oscillations. PF layer, in addition to propagation of oscillation, integrates exteroceptory signals, and MN layer integrates proprioceptory signals as well as innervates muscles.

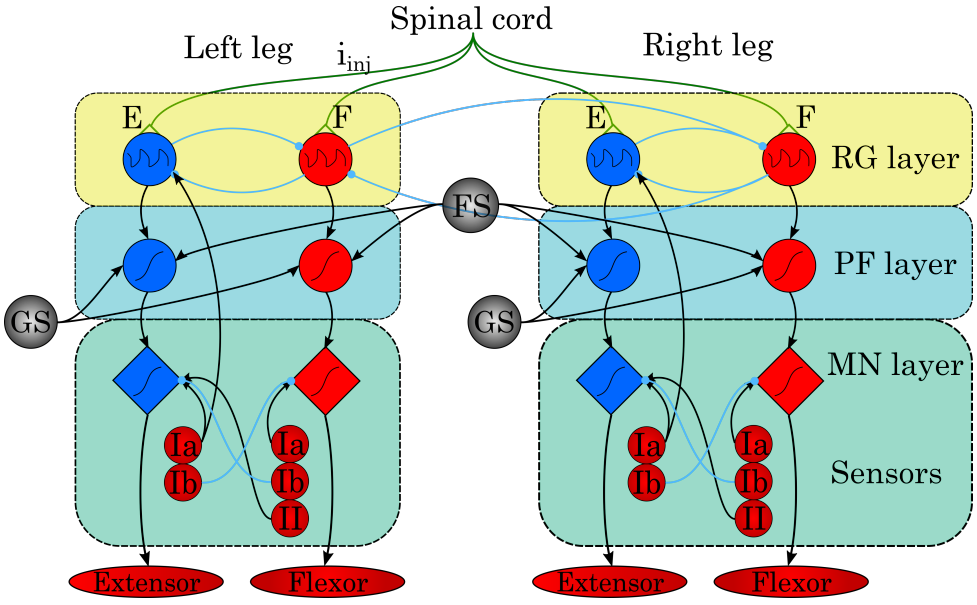
2.3.2 Rhythm Generator layer

The oscillating neuron model originates from Rowat and Selverston mesoscopic coupled neuron (Eq. 2.4). It can be rewritten as a generalized Van der Pol oscillator to support Hebbian learning rule [72]. As well as two additional rules, mentioned further. Two inhibitory coupled neurons form an oscillating unit (RG layer) of CPG.

With integrated frequency learning rule and in Van der Pol form, the equations for both half-centers are as follows (Eq. 2.7). This version has been used to control simple muscular gait and vary it's speed [90].



(a) Biological model of two coupled locomotor CPG by Rybak et al. Extracted from [58].



(b) Coupling of two legs in our CPG model.

Figure 2.3: Coupling of two leg CPGs.

$$\begin{cases} \dot{V}_F = y_F + \varepsilon \cdot i_{injF} - W_o V_E; \\ \dot{y}_F = \dfrac{y_F}{\tau_m}\left(\sigma_f - \dfrac{\tau_m}{\tau_s} - 1 - \sigma_f \tanh^2\left(\dfrac{\sigma_f}{A_f} V_F\right)\right) - \dfrac{1+\sigma_{sF}}{\tau_s \tau_m} V_F + \dfrac{A_f}{\tau_s \tau_m} \tanh\left(\sigma_f \dfrac{V_F}{A_f}\right); \\ \dot{\sigma}_{sF} = +2\varepsilon \cdot i_{injF} \sqrt{\tau_m \tau_s (1+\sigma_{sF})} \dfrac{y_F}{\sqrt{V_F^2 + y_F^2}}; \\ \dot{V}_E = y_E - \varepsilon \cdot i_{injE} - W_o V_F; \\ \dot{y}_E = \dfrac{y_E}{\tau_m}\left(\sigma_f - \dfrac{\tau_m}{\tau_s} - 1 - \sigma_f \tanh^2\left(\dfrac{\sigma_f}{A_f} V_E\right)\right) - \dfrac{1+\sigma_{sE}}{\tau_s \tau_m} V_E + \dfrac{A_f}{\tau_s \tau_m} \tanh\left(\sigma_f \dfrac{V_E}{A_f}\right); \\ \dot{\sigma}_{sE} = -2\varepsilon \cdot i_{injE} \sqrt{\tau_m \tau_s (1+\sigma_{sE})} \dfrac{y_E}{\sqrt{V_E^2 + y_E^2}}. \end{cases} \tag{2.7}$$

where F and E denote flexor and extensor half-centers. V, the membrane potential, has been used as output of neuron. y is slow membrane current, i_{inj} is the sine injected current to learn from, ε is a gain for i_{inj}, W_o is a weight of inhibitory connection between half-centers, τ_m is membrane time constant, τ_s is activation time constant for the slow current, σ_f is dimensionless shape parameter for the current–voltage curve, σ_s is the potassium conductance normalized to leak conductance, it influences the frequency of neurons, A_f is a parameter that influences the amplitude of V.

Note the $\pm$ signs of i_{inj} in-between half-centers. This difference acknowledges the fact that RG learns not only the frequency of the i_{inj}, but also its phase. Thus, to support phase difference between half-centers during learning, we need to maintain it in i_{inj}, which in case of sine, simply inverts it with -.

The constants had the following values: $\varepsilon = 1.0$, $\tau_m = 0.5$, $\tau_s = 10.0$, $A_f = 1.0$, $\sigma_f = 3.0$, $W_o = 0.2$. The initial values were: $V_F = 1.0$, $y_F = 6.0$, $V_E = -1.0$, $y_E = -6.0$, $\sigma_{s\{F,E\}} = 50.0$.

Further on, applying the work of Jouaiti et al. [72], we have added two more learning rules, changed the coupling of half-centers, and switched to y as neuron output instead of V. Eq. 2.8 shows only flexion half-center equations. Extension ones follow the system in Eq. 2.7, along with used values' symbols.

$$\begin{cases} \dot{V}_F = y_F + \varepsilon_F \cdot i_{injF} - W_i \dfrac{y_F}{1 + e^{-\rho_i y_E}} - W_c \dfrac{y_F}{1 + e^{-\rho_c y_{Fc}}}; \\ \dot{y}_F = \dfrac{y_F}{\tau_m}\left(\sigma_f - \dfrac{\tau_m}{\tau_s} - 1 - \sigma_f \tanh^2\left(\dfrac{\sigma_f}{A_{fF}} V_F\right)\right) - \dfrac{1+\sigma_{sF}}{\tau_s \tau_m} V_F + \dfrac{A_{fF}}{\tau_s \tau_m} \tanh\left(\sigma_f \dfrac{V_F}{A_{fF}}\right); \\ \dot{\sigma}_{sF} = +2\varepsilon_F \cdot i_{injF} \sqrt{\tau_m \tau_s (1+\sigma_{sF}-\sigma_f)} \dfrac{y_F}{\sqrt{V_F^2 + y_F^2}}; \\ \dot{A}_{fF} = -\mu\left(\left(\dfrac{\nu \cdot \sigma_f \cdot V_F}{A_{fF}}\right)^2 - i_{injF}^2\right); \\ \dot{\varepsilon}_F = \lambda \cdot \tanh(\xi \cdot i_{injF}) \cdot \left(1 - \varepsilon_F^2 \cdot i_{injF}^2\right). \end{cases} \tag{2.8}$$

where, in addition to Eq. 2.7, two more learning rules has been added. Amplitude learning with μ is a learning step and ν is a scale factor. Similarly, for frequency learning step adaptation, λ is a learning step and ξ is a gain for scaling the $\tanh(\cdot)$ argument.

The constants values were changed to better support rather low frequency of human gait, about 0.5 to 1.25 Hz [91]: $\tau_m = 0.35$, $\tau_s = 3.5$, $\sigma_f = 3.0$, $\mu = 10^{-6}$, $\nu = 20.0$, $\lambda = 1.0$, $\xi = 1.0$, $\rho_i = 0.1$, $\rho_c = 0.1$, $W_i = 0.124$, $W_c = 0.16$. The initial values were: $V_{\{F,E\}} = 0.0$, $y_F = 4.24$, $y_E = -4.24$, $\sigma_{s\{F,E\}} = 50.0$, $A_{f\{F,E\}} = 1.54$, $\varepsilon_{\{F,E\}} = 0.821$. The initial values were found with optimization procedure, as described in Chapter 5.

2.3.3 The other layers

The activation function of the rest of neurons is sigmoid, whose main parameters are amplitude (gain) and saturation (center shift). These neurons were previously used by Nassour et al. (see [82] for additional details). Basically, sigmoid neurons support descending control as suggested by biological evidence [87].

2.3.3.1 Sensory Neuron layer

The most numerous neurons in our controller are Sensory neurons (SN), both proprioceptive and exteroceptive, which provide feedback for the controller. The proprioceptive neurons sense changes from inside the body, such as joint or muscle values, while the exteroceptive neurons react to environmental changes such as body angle or ground reaction force. SN are the simplest of all neurons as they have only one input (Eq. 2.9).

$$SN = \frac{1}{1 + e^{\alpha \cdot (\theta - \varphi)}} \tag{2.9}$$

where SN is an output value, α is the slope value, θ is the shift of sigmoid's center, φ is an input to SN neuron of different nature.

Sigmoid function has two internal parameters, α and θ to tune curve's slope and center position respectively. These pairs for each neuron are optimized independently, but for sensory neurons we have found that there is a more handy way to control both minimum value and slope with $\theta = \alpha_0/\alpha + \theta_0$, where θ_0 is constant start of curve and α controls only the slope as on Fig. 2.4. The value of $\alpha_0 = 6$ was found empirically.

With this substitution, it is easier to define the minimal sensed signal value with θ_0 and then to tune both slope and maximum sensed value with α.

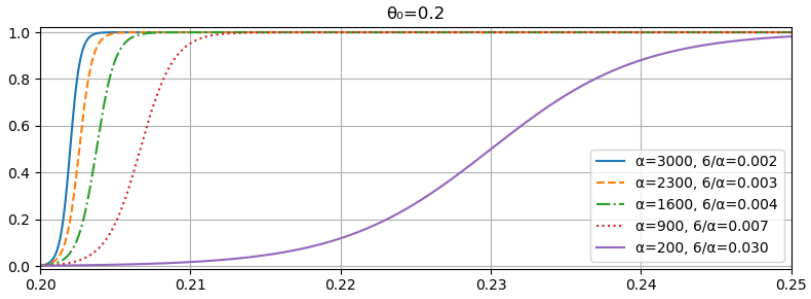


Figure 2.4: Sigmoid neuron curve with independent α, dependent θ, and constant θ_0

2.3.3.2 Pattern Formation layer

Patterns generated by RG layer are shaped by sigmoid PF neurons (Fig. 2.2b), which have 3 functions. Firstly, sigmoid PF neurons non-linearly transform RG signal range to $[0; 1]$ (Fig. 2.4). Secondly, they integrate exteroceptive signals from ground and body sensory neurons.

And lastly, they choose the domination rhythm for a CPG (flexion/extension) by partially clamping their output from upper controller. They allow of rhythm deletion of RG layer without resetting its

phase. This means that PF neuron can completely clamp their output to motoneurons while RG layer continues to oscillate. Specifically, upper signals from MLR are: α_{MLR} and θ_{MLR}. α_{MLR} changes step length and θ_{MLR} clamps further propagation of signal, therefore slowing or stopping the gait.

Each PF neuron regardless of the number of inputs (n) is defined by Eq. 2.10.

$$\begin{cases} PF = \dfrac{1}{1 + e^{\alpha \cdot \alpha_{MLR} \cdot (\theta + \theta_{MLR} - I)}} \\ I = \dfrac{W_{RG \to PF} \cdot RG + \sum_j^n W_j S_j}{n+1} \end{cases} \tag{2.10}$$

where PF is an output value, α is a value that denotes the slope of sigmoid function, θ is horizontal shift of sigmoid's center (threshold), I is an average input to PF neurons, α_{MLR}, θ_{MLR} are parameters from upper controller (MLR), $W_{RG \to PF}$ is weight of the synaptic connection between CPG neurons (e.g. RG to PF); W_j, S_j are weights and activation signals from exteroceptive neurons.

Descending control and the modulation of the threshold from the high-level controller drives the rhythm domination and deletion to the next layer.

2.3.3.3 Motoneuron layer

MN layer neurons locally control the muscles with input from previous PF layer and proprioceptive SN. MN also use sigmoid activation function and is calculated almost the same (Eq. 2.11).

$$\begin{cases} MN = \dfrac{1}{1 + e^{\alpha \cdot (\theta - I)}} \\ I = \dfrac{W_{PF \to MN} \cdot PF + \sum_j^n W_j s_j}{n+1} \end{cases} \tag{2.11}$$

where notation is the same as Eq. 2.10.

2.4 Controller architecture

The four layers: Rhythm Generator, Pattern Formation, Motoneuron, and Sensory Neuron form a CPG unit (Fig. 2.2b), which is mediated by an upper control unit (Fig. 2.5). Upper control unit initiates/stops locomotion and changes its mode, e.g. walk/run. Several interconnected CPGs form a network (circuitry), that control muscles in a musculoskeletal simulator. The simulator is described in Chapter 3.

To close the control loop, the simulator gives feedback signals, measured by sensory neurons (SN) in CPG. The set of proprioceptors, defined in Section 2.0.2, has been connected differently in each CPG network. Generally, SN *Ia*, which react to velocity of muscle contraction, were directed to limit angle and speed of joint. SN *Ib* react to force produced by muscle and improve sequencing of muscle activation or provide positive feedback for ankle balance, as described in Section 2.5. Lastly, SN *II*, presented only in flexor muscles (Fig. 2.2b), react to muscle length and also help maintaining cycle of gait or limit SN *Ib* in balance controller.

CPG model was supplemented with additional interneuron connections, according to biological evidence [88, 54]. This resulted in several variations of CPG, for each control target according to CPG network architecture (see Chapter 5). Generally, different CPG vary in exteroceptory input and coordination coupling of CPGs from different levels.

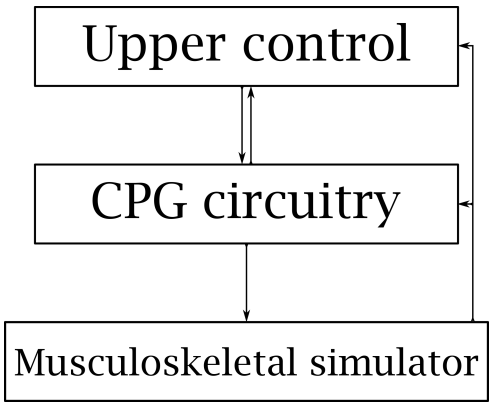


Figure 2.5: Control scheme

The resulting model is a dynamic closed-loop controller with physical simulator able to produce stable rhythmic gait in sagittal plane.

2.5 Equilibrium control

During the development of CPG controller network, we have faced the need to define roles of *Ib* and *II* afferents. We actively use load receptors and reflex loops in equilibrium control. Biological evidence suggests that these afferents form simple reflexive loops and participate in balance control [88, 54, 89, 44]. So we have decided to separate them into a reflex controller for maintaining balance that acts alongside the CPG controller.

Song et al. in [89] have presented a spinal controller to produce gait using only afferent feedback and no oscillators. We have adopted some of their reflexive loops. Generally, loops that are active during stance phase and not entrainment during walk.

The resulting reflex controller is capable to maintain balance of the musculoskeletal model for several seconds at the start of a simulation on its own. Unfortunately, the controller does not allow the model to keep balance during gait and to remove the elastic harness (see Section 3.3.2.1).

2.5.1 Reflex controller

We have adopted stance modules for reflex loops from the work of Song et al. [89], named M_1, M_3, and M_5. Generally, the work has ten modules for the whole gait cycle with six swing-phase related modules that create entrainment for gait. They were not adopted as we use CPG controller for this purpose. As well as module M_2 that prevents knee over-extension as musculoskeletal simulator does this for us.

Each module has a defined function that allows to independently implement and test each of them.

Fig. 2.6 shows all adopted modules. Module M_1 realizes compliant leg. This is implemented as positive feedback of muscle force in extension muscle groups in each joint. This module is activated by load on the leg and keep the leg extended during stance phase. Also, M_1 implements the reflex of standing on toes during fall forward as a last resort in keeping balance.

Module M_3 balances trunk in hip joint using body angle exteroceptor. Basically, when body is inclined

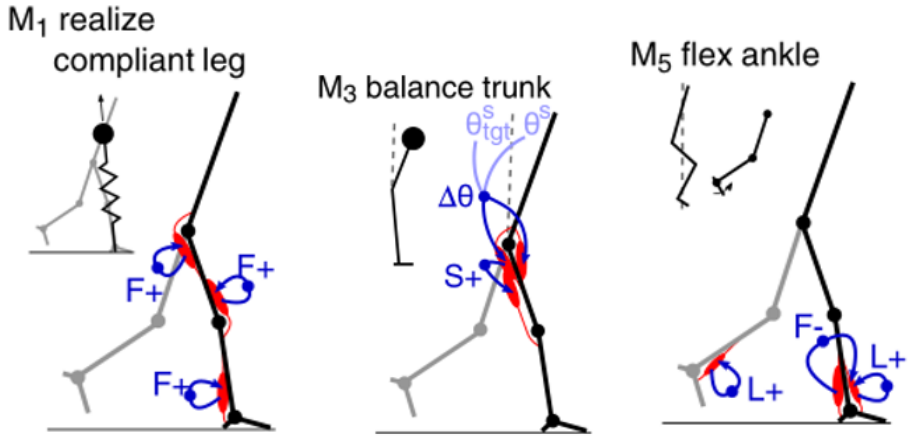


Figure 2.6: Used reflex modules. Extracted from [89].

forward, the module excites extension hip muscles. And vice-versa, when the body falls back, the flexion hip muscles are activated to bring the body in upward position.

We have supplemented this module with ground reaction reflexes in hip muscles to balance the center of mass during changes in feet pressure. Namely, ground sensor in heel excite hip flexion muscles and inhibits hip extension ones. And toe pressure sensor excite hip extension muscles.

Module M_5 controls ankle joint during stance. It uses both muscle force and length afferents to counter the excessive activation of ankle flexion in M_1 and generally keep balance with ankle joint.

When the stance balance cannot keep up and the fall is inevitable, the model should perform a single step, which means using reflex and CPG control simultaneously. This integration requires simultaneous training of both controllers and is currently in progress with the latest version of CPG network.

2.5.2 Arm control

Humans actively use their arms to supplement balance control during gait. This behavior has been researched and modeled with CPGs [88, 44] and in open loop with dynamic musculoskeletal model [92].

We have implemented a simple arm control during gait to help maintain balance. As our musculoskeletal model does not have upper-body muscles, the control is made with PID controllers that produce shoulder joints' torque based on three values and hip joints' angle. More detailed in Section 3.3.1.4.

2.6 Conclusion

In this chapter, we have described the concept of central pattern generator, its biological evidence, current models, and application in bio-inspired control, including robots. We have described the features of our model of CPG network as well as its functioning and composition. And, finally, we have mentioned the other control components, relevant to gait and balance.

Next chapter describes the used musculoskeletal simulators and their features. Along with the details of controlling them with CPGs.

Chapter 3

Musculoskeletal model of human body

The target for our neuronal control of gait is a musculoskeletal model of a human (Fig. 2.5). The CPG network forms excitation signals for the muscles and expects afferent feedback from muscles and environment.

Numerical simulations of complex physical interaction in-between human body parts and with environment requires a skeletal model. Such models of varied rigor provide researchers with biological, bio-mechanical, dynamic, kinetic, and kinematic data that would be impossible to obtain or complicated to gather otherwise. This data include time-varied body acceleration, joint moments, centers of mass and pressure, etc. Simulated data help to evaluate the motion coordination and applied forces of a patient with augmentations or predict the skeletal system's state after a surgical intervention.

Further increase of bio-mechanical accuracy of human locomotor system simulations lays in adding of muscular system. Muscles create joint torques that act on the skeletal system. Muscular part complements skeletal model into musculoskeletal model that realize muscle excitation into body-environment interactions.

Existing musculoskeletal models can simulate realistic locomotion using a priori calculated muscle excitation data. Excitation is obtained by inverse dynamic algorithms using motion capture records of kinematic and dynamic human data, but only in open-loop [4, 92, 93], i.e. without any feedback coming from the environment or the musculoskeletal system itself. Closed-loop simulation requires to consider physiological and environmental feedback and a controller that varies its output depending on this feedback.

To simulate a musculo-skeleton system, one could start from a cellular level [94], but that would be an overdoing as the aim of this work is not the most realistic simulation of human lower limbs; any simulator that uses classic mechanical Hill muscle model [95] will do. Such model, based on work of Thelen [96] is implemented in OpenSim[2] or in GAIT2DE based on the work of [97, 3], whose Hill muscle is based on the work of Zajac [98].

In the first part of this work, GAIT2DE simulator has been used. Then, a more anatomically advanced model implemented in OpenSim was used.

[2] `http://opensim.stanford.edu`

3.1 GAIT2DE, a simple musculoskeletal model

Human locomotor system simulator "GAIT2DE"[3] simulates muscle activity and its influence on skeleton to actuate legs movements in sagittal plane taking in account internal and external dynamic effects. The human locomotor model consists of 7 body segments and 16 muscles (Fig. 3.1). It is implemented as MATLAB MEX function. MEX function's source code was available[4], but it results from a code-generation tool whose input instructions are unavailable.

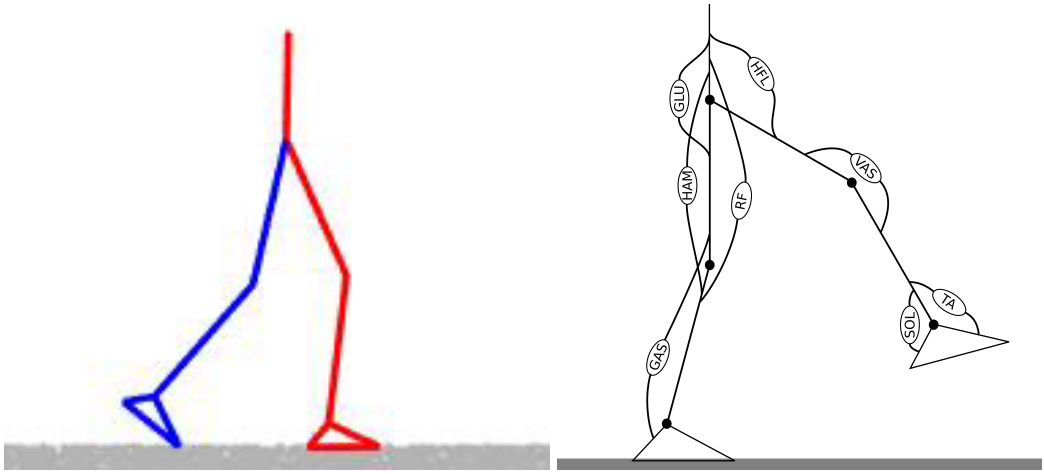


Figure 3.1: GAIT2DE model. Left: default MATLAB visualization. Right: structure of the model with muscles of one leg.

The model has nine kinematic degrees of freedom, seven body segments, eight muscles per leg, and its dynamics and outputs are twice differentiated with respect to all inputs. The model accounts for physical phenomena (ground friction, external forces, dynamics of limbs, etc.).

Body segments are and thigh, shank, foot in each leg accordingly connected by joints of hip, knee, and ankle. Each of body segments has the following parameters: mass, length, center of mass, moment of inertia of a male with body mass of 75 kg and body height of 1.8 m. Eight muscles in each leg are HFL, Gluteus, Hamstrings, Rectus Femoris, Vastus, Gastroc, Soleus, and Tibialis Anterior. Other properties are described in [97].

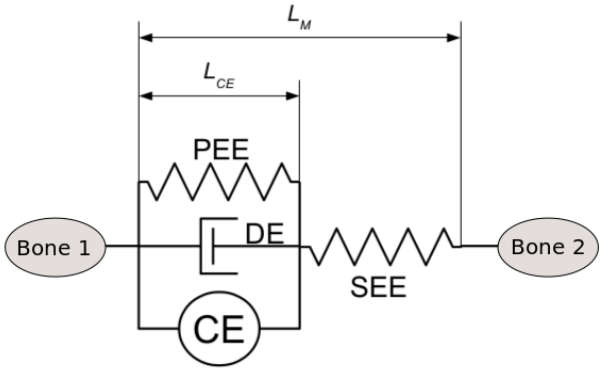
3.1.0.1 Muscle model

This simulator implements the well-known Hill muscle model [95] that consists of parallel elastic element (PEE), damping element (DE), series elastic element (SEE), and contractile element (CE) (Fig. 3.2a). The CE element is controlled by neural excitation. The activation function of muscles is of Eq. 3.1.

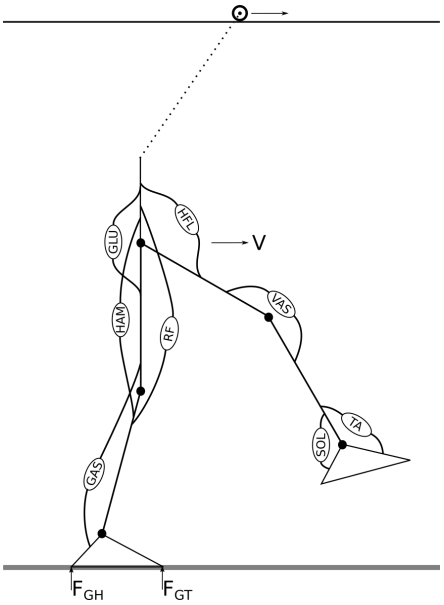
$$\frac{da(t)}{dt} = (u(t) - a(t)) \cdot \left(\frac{u(t)}{T_{act}} + \frac{1 - u(t)}{T_{deact}} \right) \tag{3.1}$$

[3] Developed by Antonie van den Bogert, Orchard Kinetics LLC.
[4] `http://www.orchardkinetics.com` is currently taken down.

where $a(t)$ is an active state of a muscle, $u(t)$ is input neural excitation, $T_{act} = 10$ms and $T_{deact} = 40$ms are time constants for activation and deactivation. For values of parameters, see [96].



(a) Hill muscle model (extracted from [97] and modified). L_{CE} is length of contractile element and L_M is full muscle length.



(b) Body segments, joints, and muscles of GAIT2DE model. Muscles are drawn for one leg.

Figure 3.2: Details on GAIT2DE model.

The force in CE—or muscle fiber—depends on its active state, length, and lengthening velocity. The linear DE ensures that muscle model is solvable when the force in the CE is at or outside the asymptotic values of the force-velocity relationship. PEE and SEE represent passive properties of muscle fibers and surrounding tissue.

3.1.0.2 Modifications to GAIT2DE

This musculoskeletal model was modified in order to control the muscles by the signals generated by a CPG circuitry and obtain required feedback.

The first modification was to divide a single muscle force on each foot into two forces on heel and toes. This was an easy change as the forces were internally calculated apart and added together.

Also a sliding support elastic to the model that acts as a harness to help maintain vertical position has been added (see Fig. 3.2b). Elastic moves in horizontal directions to approach a position between feet while maintaining constant height of about 1.7 m. The elastic was implemented utilizing GAIT2DE's input for external forces. The external force F is calculated as in Eq. 3.2.

$$F = -k \times (d - L_0) - F_p/2 \tag{3.2}$$

where $k = 5\ kN/m$ is stiffness of elastic, d is distance to attaching point, $L_0 = 0.7\ m$ is resting length of elastic, and F_p is force from last time frame for simulation of elastic's energy dissipation.

This addition was required as CPG controller generates only rhythmic movements and doesn't deal with reflexes and balance control on its own. Section 2.5 introduces a reflexive balance controller, which requires a fair bit of tuning and integration with CPG controller, before it can be used for balance.

The modifications and overall structure of the model are shown on Fig. 3.2b. F_{GH} and F_{GT} are ground reaction forces on heel and toe, V is velocity of pelvis. Dotted line is supportive elastic to help dealing with equilibrium issues.

Using the coordinates and angles of joints from previous state along with muscle excitation vector and external forces, the simulator returns joint spacial and angular velocities and ground reaction and muscle forces. Simulation setup and results with this musculoskeletal model are described in Chapter 5.

GAIT2DE has proven itself sufficient during initial development of CPG controller, but it lacks anatomical details to display the possibilities of the controller. GAIT2DE only has 16 muscles, while human legs contain about a hundred of them. Additionally, the inner anatomical structure of this model is not available for review.

3.2 Musculoskeletal model in OpenSim

For further development the simulation we have used framework OpenSim [5], developed by Stanford university. OpenSim[5] is a freely available tool for modeling and simulation of movement. It is used in rehabilitation, orthopedics, robotics, ergonomics, performance, and wearable device design. But, most importantly, OpenSim has been used for implementation of musculoskeletal model of human by Rajagopal et al. [4] (Fig. 3.3).

This model has 37 degrees of freedom (DoF), 22 body parts, 80 Hill-type muscles that control 8 leg joints and 17 torque drives that control 7 joints of the rest of the body. The authors have developed it on the basis of anatomical measurements of 21 cadaver specimen and magnetic resonance images of 24 young healthy subjects. The model has been verified by running forward-time simulation and comparing results to electromyographic and dynamic data of healthy running and walking.

By using Hill muscle model [99] with anatomical skeletal geometry, the authors achieved a great reduction of calculation time, while keeping acceptable accuracy of muscle simulation. The original model was a priori scaled to a person using OpenSim built-in tools.

OpenSim is implemented in C++ language. The developers also provide two options to use the framework with scripting languages, MATLAB and Python. Further on, Python wrapping over OpenSim is utilized, extracted from the "osim-rl" project[6] for the initial insight to scripted OpenSim.

3.3 Modifications to the musculoskeletal model

The original musculoskeletal model has been made for inverse dynamics calculations, thus lacking a few features needed for this project. Specifically, a contact with ground and joint angle limits. Rajagopal et al. [4] have computed muscle excitation from motion capture data, thus implicitly dealing with issues of joint angle exceeding anatomical limits and contact with ground.

[5] `https://opensim.stanford.edu`
[6] `http://osim-rl.stanford.edu`

In this thesis, we have modified the model with mentioned features and a few more to adapt it for my research purpose, which is forward-time closed-loop gait simulation. The final looks of the model are shown on Fig. 3.4.

Model's modifications can be divided into two groups:

I. Simplifications of the model:

1. Restricting the model translation outside the sagittal plane;
2. Reducing the number of muscles;
3. Disabling back and wrist joints;
4. Removing all, but 2 shoulder torque-actuators;

II. Extensions to the model:

5. PID control of shoulder joints;
6. Supportive harness;
7. Limitations of joint angles;
8. Contact geometry of feet and ground.

Figure 3.3: Musculoskeletal model in OpenSim by Rajagopal et al. [4].

3.3.1 Simplifications of the model

The following simplifications remove the unused components of the model to decrease computation time of simulation. Computations with OpenSim and Python interface are made in a single CPU thread, thus profiting from simple restrictions of the model.

3.3.1.1 Restricting the model translation

Four restrictions of model's movement were made to simplify the task of gait simulation in the sagittal plane. Our goal is to simulate straight-line human gait and the effects of pelvis non-sagittal displacement had been considered negligible.

We have disabled four coordinates: Y-rotational axis of whole body (`pelvis_rotation`), Z-translation axis of whole body (`pelvis_tz`), X-rotational axis of the joint (`hip_adduction`), and Y-rotational axis of the joint (`hip_rotation`). The axes in OpenSim are as follows. X-axis is forward/backwards, Y-axis is up/down, and Z-axis is right/left.

3.3.1.2 Reducing the number of muscles

The original model contains 40 muscles on each leg. Previous modification is closely connected to the second one. The set of simulated muscles is reduced to both lower the computation time and to remove the muscles that don't contribute to movement in sagittal plane.

OpenSim can't simulate "wide" muscles and they need to be divided into several single-fiber muscles. Such as each of "triangular" Gluteus maximus, Medius, and Minimus muscles is implemented as three independent muscles arranged as a triangle.

We are not using all the muscles that the model offers, only the following 28 muscles of each leg:

- With Pelvis-Femur connection: Adductor Magnus (Ischiocondylar portion), Gluteus Maximus (3 items), Iliacus, Psoas, Sartorius.
- With Pelvis-Tibia connection: Biceps Femoris (Long Head), Gracilis, Tensor Fascia Latae, Rectus Femoris, Semimembranosus, Semitendinosus.
- With Femur-Tibia connection: Biceps Femoris (Short Head), Vastus intermedius, Vastus lateralis, Vastus medialis.
- With Femur-Calcaneus connection: Gastrocnemius (Lateral part), Gastrocnemius (Medial part), Peroneus Brevis, Peroneus Longus, Tibialis Posterior.
- With Tibia-Calcaneus connection: Soleus, Tibialis Anterior.
- With Tibia-Toes connection: Extensor Digitorum Longus, Extensor Hallucis Longus, Flexor Digitorum Longus, Flexor Hallucis Longus.

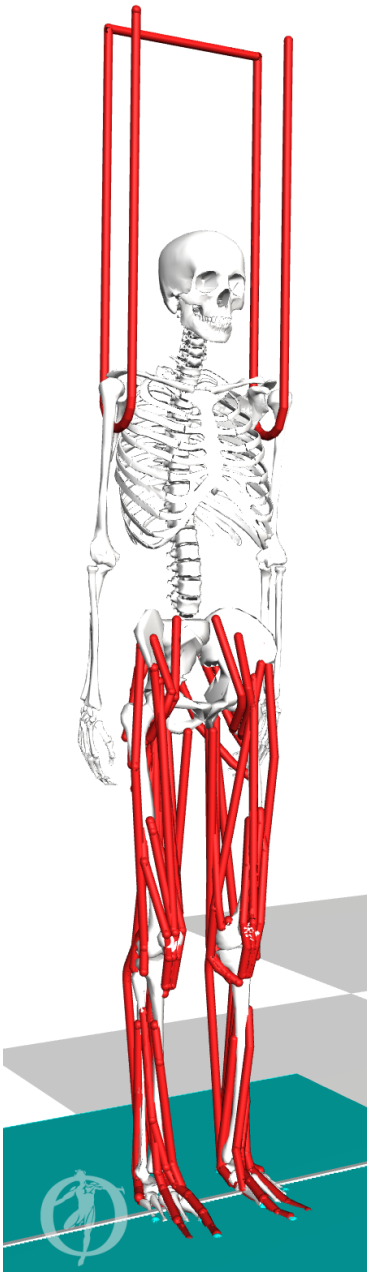


Figure 3.4: Musculoskeletal model with modifications.

3.3.1.3 Disabling back and wrist joints

The back joint and both wrist joints are redundant to the simulation goal. The back joint in the model is located between the pelvis and lower part of spine and contains 3 DoFs. Due to this reason, they were disabled, e.g. set in default position.

Similarly, wrists were set to more natural walking position as shown on Fig. 3.4. Each of them contained 2 DoFs.

3.3.1.4 Shoulder torque-actuators

Originally, the model contained 17 idealized torque-actuators to control the upper body, which doesn't have any muscles on it. For simplicity, 15 of them were removed along with all of the marker positions.

Two utilized actuators control Z-rotational DoFs of shoulders. Shoulder control was implemented to simulate more natural movement by counterbalancing leg movements, as suggested by biological evidence [88].

Each shoulder actuator is enabled by a PID controller[7], with coefficients $P = 0.7$, $I = 0$, $D = 30$. These coefficients were chosen manually to emulate the hypothetical CPG connection between arms and legs, when arm obtains a little push, depending on the contralateral hip position and velocity. The PID calculates the torque for the shoulder from its current and target positions.

The target position is selected among 3 values ($0°$, $\pm 10°$) based on the position and speed of the opposite hip joint. The algorithm is as follows.

[7] from `https://pypi.org/project/simple-pid`

- PID target= 0° when opposite hip moves towards the 0°.
- PID target= −10° when opposite hip moves backwards while behind the body.
- PID target= +10° when opposite hip moves forward while in front of the body.

Figure 3.5 shows the signals of PID regulators and relevant joints during gait. The arm moves along to the opposite hip.

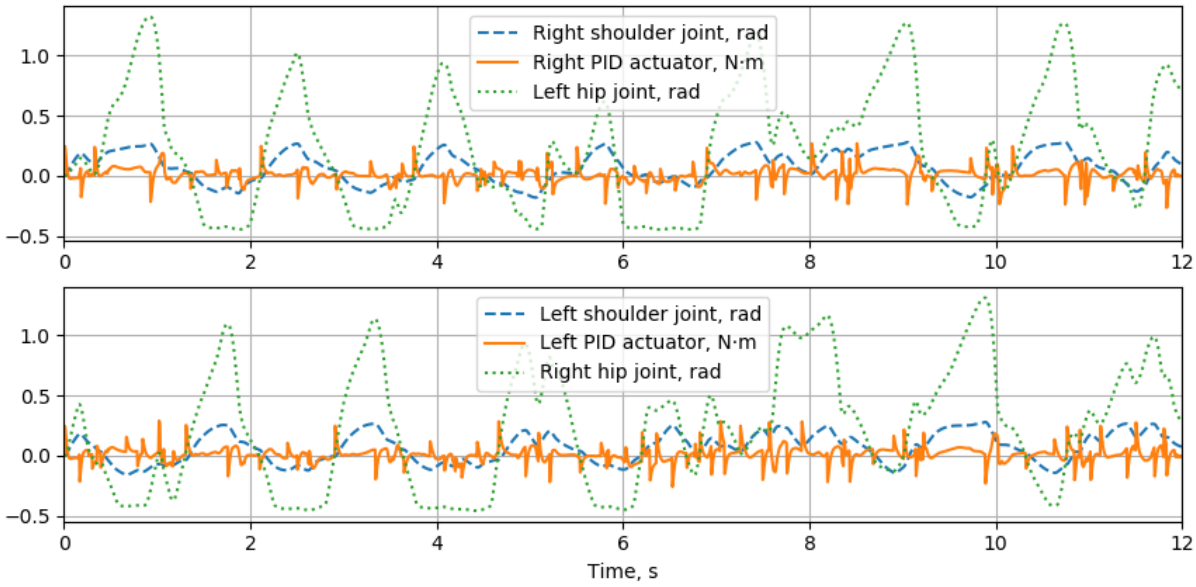


Figure 3.5: Signals from shoulder's PID controllers.

3.3.2 Extensions to the model

Several additions to the model have been made to accommodate it to our research. Like with GAIT2DE model, wWe have added a supportive harness-elastic to prevent body from falling due to the insufficient vestibular system capabilities in current work. An explicit limitation of leg joint angles is implemented to bring their anatomical capabilities to real ones with correct behavior of the joint with soft tissues and bones. Moreover, contact spheres are added to the feet and a ground half-space that interacts with the spheres.

3.3.2.1 Supportive elastic harness

The harness is modeled as an additional muscle. It supports the model's upward position by wrapping around half-cylinders at model's shoulders (Fig. 3.6) and pulling up to virtual rails above model. The top of harness moves along the model's center of mass (CoM) in the horizontal plane. It prevents the model from falling as the simulation of vestibular system is not sufficient. It can be considered as an equivalent to real harness for patients used in hospitals.

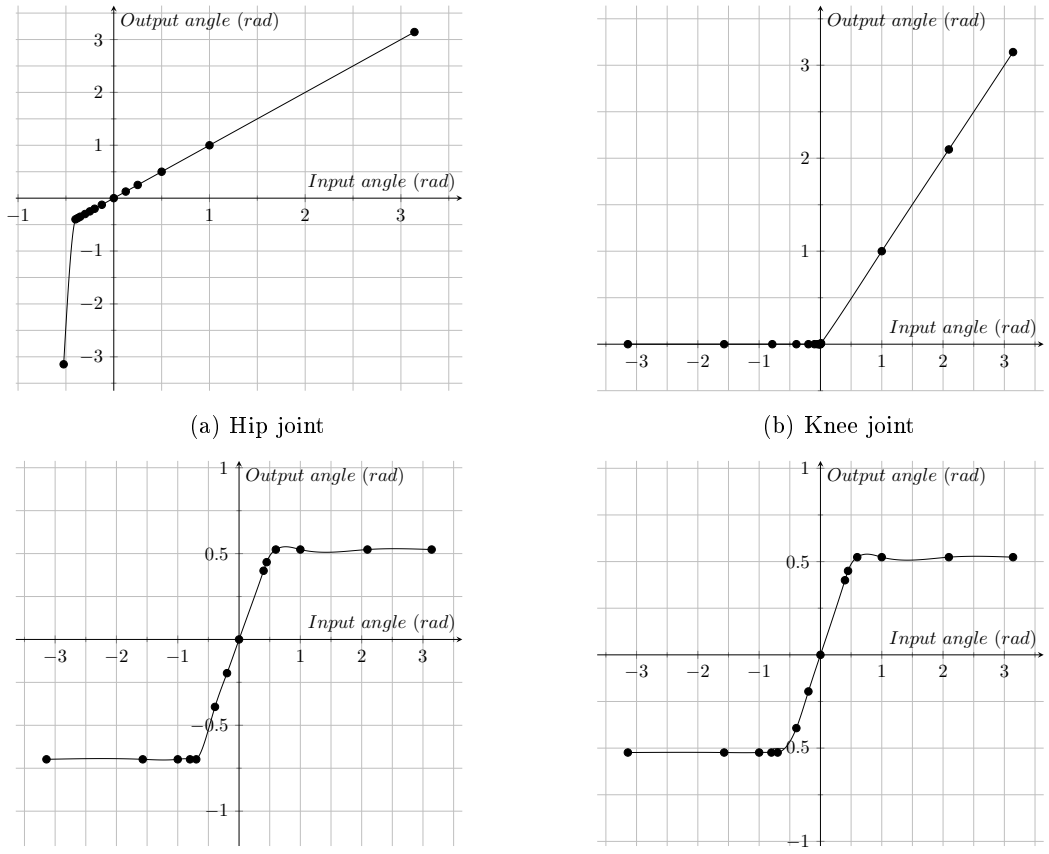
Figure 3.6: Connection of the elastic harness.

3.3.2.2 Limiting joint angles

The original model is intended to be used with a priori calculated muscle excitation. As a test, model's authors [4] have used Computed Muscle Control that tracks the input kinematics from motion capture to calculate muscle excitations.

Thus, they didn't need to prevent joint rotation beyond anatomical capabilities as the muscle excitation was pre-calculated to account for that. But, as there's no simulation of bones interacting with each other and with soft tissue and skin, it is nessesary to implement the limitation to joint angles.

Joint angle limitations are achieved by assigning a transformation function for each considered degree of freedom. First, the joint type needs to be changed to `<CustomJoint>` with a `<SpatialTransform>` function in OpenSim's model file `*.osim`. The function is `<SimmSpline>`, like shown on Fig. 3.7 for each joint.



(a) Hip joint

(b) Knee joint

(c) Ankle joint

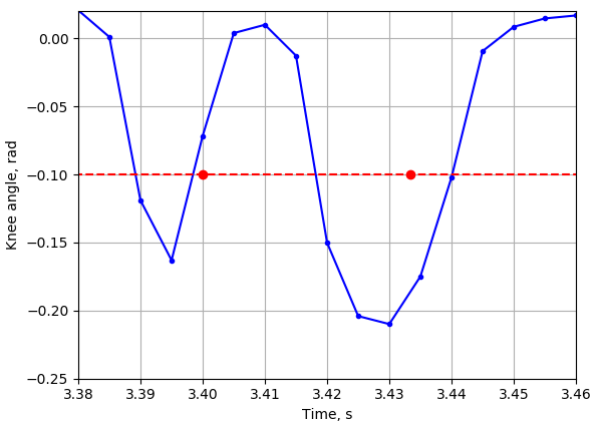
(d) Metatarsophalangeal joint

Figure 3.7: Spline functions to limit joints.

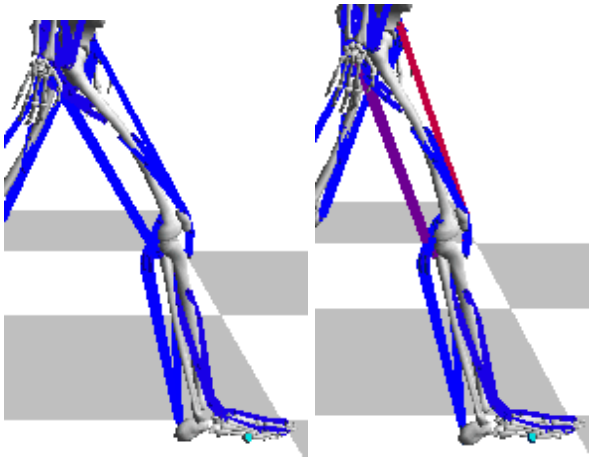
This `<SpatialTransform>` (in this case it's angular) function is applied after calculating the physics-based angle of a joint and it's influence on the state of the model, but before rendering. Utilization of this function reveals two side-effects.

Data obtained from the model through `coordinate.getValue(state)` shows first side effect: joint turns beyond limited angles, which is not observable on rendered body simulation. Fig. 3.8 shows this

mismatch. Fig. 3.8a shows that knee angles at frame times of 3.4 and 3.433 seconds have values of -0.07 and -0.177 radiant (over-extension). But rendered frames on Fig. 3.8b clearly indicate positive (flexed) and zero angles. It is worth noting that the rendering in OpenSim is hard-coded at 30 frames per second.



(a) Fragment of a plot of right knee angle with negative angle values. Dots on red dashed line denote frame timings.



(b) Two frames of model rendered at 3.4 and 3.433 seconds respectively.

Figure 3.8: Mismatch between joint angle plot and rendered frames during gait simulation with limited joint.

The second side-effect is less clear and may have a better solution. Describing only considered (sagittal) degrees of freedom of joints with transforming spline (or polynomial) functions leads to integration errors. The other DoFs were assigned with constant zero values. Our solution is to assign one of non-sagittal DoFs of each joint a spline function that is near zero on linear region and as small as possible (empirically) to remove errors. Our reason that can explain this side-effect suggests that the energy of sagittal movement is not dissipated with limitation and has to be directed to another DoF.

3.3.2.3 Contact with ground

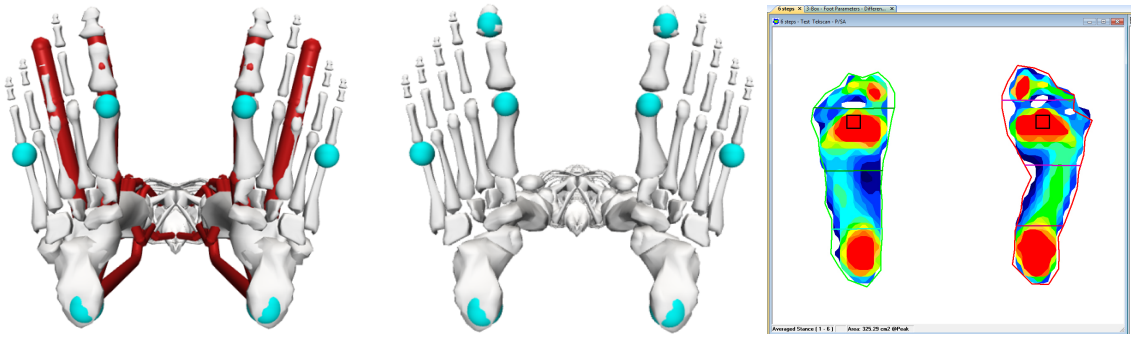
Similarly to the previous modification, model's contact with the ground was not implemented in the original simulator as it was implied in the movement tracking data.

We propose to modify model interaction with ground through several contact spheres on feet that implement Hunt-Crossley contact model [100]. The possibility to use a mesh surface of the feet for calculating the contact was omitted due to higher calculation time and difficulty to implement. This is left for the future work.

The ground is implemented as a half-space, though it is possible to use the arbitrary surface for the ground in the future to simulate gait on uneven terrain. The number of contact spheres on feet may vary. At the initial stages of development, 6 spheres (Fig. 3.9a) and a locked metatarsophalangeal (MTP) joint are used.

However, the experiments have shown that implementation of such rigid feet is insufficient for a good ground contact and also the toe contact is very important for equilibrium. So, one more sphere on each

hallux (big toe, Fig. 3.9b) has been added according to the force profile of human foot[8] (Fig. 3.9c). Moreover, we have unlocked and limited the MTP joint (Fig. 3.7d).



(a) Initial position of 6 contact spheres on model's feet.

(b) Position of 8 contact spheres on model's feet.

(c) Pressure force profile on human feet.

Figure 3.9: Foot/ground contact of the model.

The model's sagittal movement accounts only for two contact point on each foot, so the forces of toe contacts are added together for sensory neurons.

These extensions to the model conclude descriptions of the musculoskeletal model. Next, we describe connections from CPG controller to the physical simulator.

3.4 Connecting controller to the model

In general, the connection of the CPG controller to the model consists of four steps: initialize input and output arrays, pass excitation for each muscle and actuator, integrate for Δt, get results and feedback.

As described in Chapter 2, each CPG is designed to control two mutually antagonistic groups of muscles. The composition of these groups depends on architecture of CPG controller and is given in the Chapter 5.

3.4.1 Implementation of the platform

The gait simulation platform is modular and the concrete implementation of the platform is arbitrary, being dependent on the interface of musculoskeletal simulator. Although the control loop is generally the same for both musculoskeletal simulators—pass excitation, integrate for Δt, get feedback—little differences exist and are described in this section.

Initially, GAIT2DE model [97], described in Section 3.1 has been used because this simple 16-muscle musculoskeletal model is implemented as .mex function in MATLAB [101].

Further, we have switched to a more advanced musculoskeletal model by Rajagopal et al. [4] implemented in the OpenSim simulator [5]. Moreover, OpenSim offers MATLAB and Python interfaces to access it's functionality without using C++. This model and reasons for switching are described in Section 3.2.

[8] https://www.tekscan.com/application-group/medical/foot-function

MATLAB interface for OpenSim at the time of this work was inapplicable as MATLAB was refusing to call OpenSim's `Manager.initialize()` method, as it was conflicting with MATLAB's Java method with the same name. Different ways were considered to override conflicting method, to delete that Java class, but it's important for MATLAB start up. Due to that issue, we have switched to Python language.

3.4.2 Biological feedback

In the work [102], we have been using the value of joint angle as input to sensory neuron for muscle feedback. This approach is similar to robotics and it allowed to compare modeled gaits of robot and human.

In this thesis' work, we explicitly focus to use only biologically plausible signals to control human gait on spinal level (see Section 2.0.2). Five dynamic values are used as feedback signals:

- Three proprioceptive muscle signals:
  1) muscle stretch velocity signal from muscle spindles by *Ia*-type sensory fibers;
  2) muscle force signal from Golgi's tendon organ by *Ib*-type sensory fibers;
  3) muscle length signal from muscle spindles by *II*-type sensory fibers.
- And two exteroceptive signals from environment:
  4) ground reaction forces on toes and soles;
  5) angle of model in sagittal plane as substitution of vestibular signal.

3.4.2.1 Muscle stretch velocity

Type-*Ia* afferents indeed affect spinal gait [54]. They consist of sensory fibers that connect muscle spindles with spinal cord to transduce muscle stretch proprioception. Muscle spindles are located inside of muscle body and react to change of length of muscle, adapting quickly, unlike *II*-type fibers.

In this thesis, the role of *Ia* sensory neurons (SN) is to prevent excessive joint angular velocity. The sensors measure muscle's positive stretch velocity, i.e. muscle's lengthening. And by this, SN *Ia* indirectly react to opposite muscle's excessive stretch. Upon activation, SN *Ia* participates in motoneuron (MN) excitation to prevent the joint from turning too fast. Muscle's velocity of contraction is maximal during opposing MN's impulse.

SN *Ia* reacts to positive part of stretch velocity curve only when joint velocity goes below its supposed minimum. This means that if flexion muscle lengthens too fast, then opposite extension muscle turns joint to negative direction too fast. And flexion MN accordingly counters it.

3.4.2.2 Muscle force

Type-*Ib* sensory fibers connect Golgi tendon organ (GTO) with spinal cord to transduce developed muscle force proprioception [54]. GTO connects muscle fibers and tendon body and is innervated by *Ib* sensory fibers. When muscle generates force, sensory terminals get compressed and neuron generates force signal.

SN *Ib* biological function is [14]:

- the autogenic inhibition reflex (sudden relaxation of muscle upon development of high tension);
- to affect the timing of the transitions between the stance and swing phases of locomotion;
- responsible for the stimulated entrainment (without CPG).

In this work, the role of sensory neuron type-*Ib* varies. In earlier controller models SN *Ib* inhibited opposite Pattern Formation (PF) neurons to improve sequencing of muscle activation. This is achieved by reducing or skipping opposite MN's peaks in case constant position is required.

In later models, SN *Ib* was moved from CPG to reflex equilibrium controller (see Section 2.5). There, the force signal is used in positive feedback loops to realize the leg under load.

3.4.2.3 Muscle length

Type-*II* sensory fibers connect muscle spindles with spinal cord to transduce muscle stretch proprioception. They are similar to *Ia* fibers, but are non-adaptive, thus reacting to muscle's length instead of velocity.

The role of SN *II* also varies. In early models, SN *II* were part of CPG controller. They were present only on flexion side and excited extension motoneuron MN to prolong excitation phase in walking. This includes pushing hip and ankle against ground and keeping knee straight during support phase.

In later models, SN *II* was also moved from CPG to reflex equilibrium controller (see Section 2.5). There, the length signal is used for ankle muscles to detect falls and ground slopes.

3.4.2.4 Ground reaction force

Ground sensing and pressure signals are crucial to fine locomotion control [54]. They not only contribute to gait stability due to ground irregularity, but also to gait phasing during unperturbed constant walk.

Ground reaction force is measured in three contact points on each foot (see Section 3.3.2.3 and Fig. 3.9). Reaction force on heel is used for SN GB (ground backward sensory neuron) and force for SN GF (ground forward sensory neuron) is obtained by adding contact forces on toes together.

3.4.2.5 Inclination angle of model

The angle of head relative to the ground from vestibular system is one of three inputs to balance control, along with visual information and ground sensing [56].

In musculoskeletal models, used in this thesis, the head is a single physical body together with the trunk, whose angle in global coordinates has been used for vestibular signal.

This angle is used for SN FB (fall backward) and SN FF (fall forward). SN FB reacts to positive trunk angle (when model leans backwards) and participates in exciting flexion of hip and ankle muscles to prevent falling back. Similarly, SN FF prevents falling forward by exciting extension muscles.

In controller developed for OpenSim model, vestibular signal consists of two additive components. The first component is the angle itself. The second is angular velocity with a gain $K_{dt} = 0.03$. This composition of two components act as a PD-controller (proportional-derivative). A small derivative component allow gait controller to sense the future position of body angle to react accordingly due to inevitable biological delays between sensors and control networks.

3.5 Conclusion

In this chapter, we have described the used musculoskeletal simulators, their features and required modifications to adapt them for our needs of bio-inspired CPG control in closed loop with realistic afferent feedback.

First simulator uses simple musculoskeletal model in MATLAB with 16 muscles and 7 body-parts. It is suitable for initial development of the controller, but is limited in extensions and anatomical correctness.

Second simulator uses OpenSim physical environment and model that is balanced between anatomical correctness and speed of simulation. This model is easily extendable.

The next chapter provides some means to analyze simulated gait and additional procedures that support simulations, such as evaluating the resulting gaits and optimization of controller's parameters.

Chapter 4

Simulated gait analysis

The performed gait simulations aren't very useful on their own. The results must be analyzed and compared to human gait. As a close matter, the controllers, presented in Chapter 2 have large number of parameters, which do not correspond to measurable human motor characteristics. Thus, we are left with only non-invasive, external human gait data to utilize. The controller parameters are proposed to be searched for with both manual tuning and optimization algorithms. Manual tuning is performed according to assumed parameters' roles, described in Chapter 2.

This chapter deals with the matters of simulated gait evaluation and controller parameter search by comparing simulation to real gait. First, we describe the periodic gait cycle that is of great help in decomposing gait with easy to recognize events into phases for future processing. As well as the human data itself and the ways to bring it on par with the simulated gait data.

Then, we consider the controller parameters search, used methods, and target functions. And finally, the way to evaluate the simulated gait by its similarity to the real human gait.

4.1 Gait cycle

Normal healthy human gait is assumed to be periodic and symmetric during alternating progression of legs. Therefore, one can pick a single period in gait with it's time borders.

Gait phases and events highlight important events while walking. Usually, the gait cycle start is recognized at Initial Contact (IC) event [103, 104, 105]. IC is the moment when leading limb's heel touches the ground (Fig. 4.1). The leading limb is chosen empirically and in this work it is right leg.

Then, regarding symmetry of movements of right and left leg, one can divide the gait cycle in two equal parts by a symmetric event, IC of trailing limb, or Opposite IC (OIC). That is a contact of left leg's heel with the ground. If one starts the gait cycle from IC event, then IC happens at 0% of the cycle and OIC at 50%.

Additionally, there are two more symmetric events with ground contact, Toe-Off (TO) and Opposite TO (OTO). TO is a moment when the leading limb's toes are elevated and stop contacting with the ground. Roughly, TO happens at 61% of the cycle and OTO at 11%.

These are 4 generally used events [103, 104, 105] that divide human gait cycle on 4 phases. These phases regard a type of support during gait cycle. Phase from IC to OTO is a double support phase

when both legs touch the ground. Then, the left leg transit into swing phase, the body stays in a right single support phase until the OIC phase. From OIC to TO phase is again a double support one until TO event. After TO and until the next cycle's IC, the phase is a left single support while right leg is in swing phase.

These 4 phases are widely used and are easy to identify, but they are sparse, leaving big empty zones in single support phases. There are 3 more events that divide single support phases. They are Heel Rise (HR), an event in the middle of right single stance phase when the right heel is elevated during foot's push off the ground.

Left single stance is divided by two events, Feet Adjacent (FA) and Tibia Vertical (TV). FA happens when the right leg during its swing goes near left supporting leg. After that, TV event happens when right knee joint is extended and passing the vertical position of the shin (containing tibia bone).

The result is 7 events of the gait cycle shown on Fig. 4.1 as phase names as well as other ways to divide gait cycle. The events themselves are in the middle of the phases that are divided equally in between events.

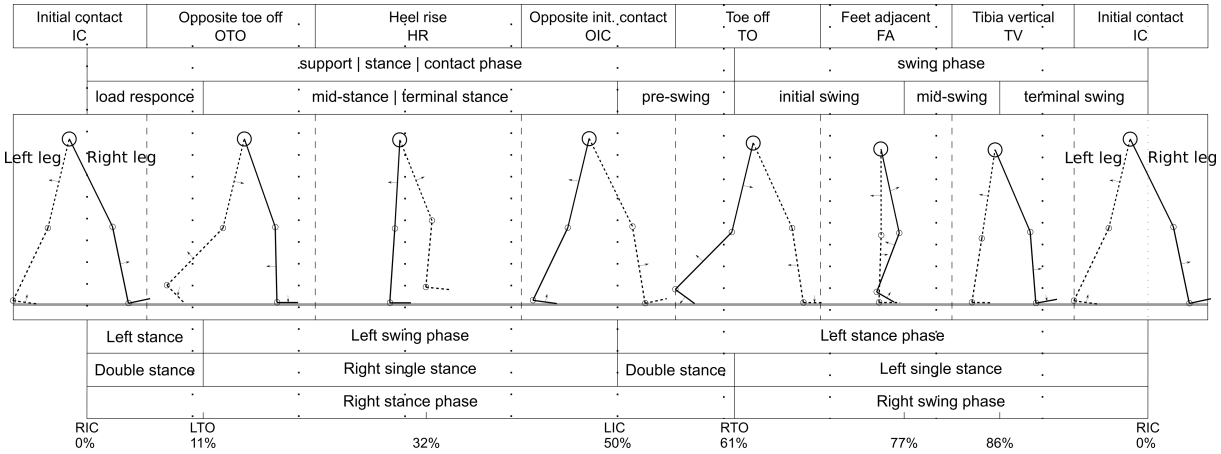


Figure 4.1: Seven phases of normal human gait cycle.

This gait cycle consisting of 7 phases is dense enough, but it's not symmetrical between legs. The right stance phase contains HR and the left one has FA and TV. This leads to obvious addition of symmetric phases to the cycle.

10-phase gait cycle

To make a default 7-phase gait cycle symmetric and relatively dense, we have added 3 opposite phases:

1. another Feet Adjacent (FA2) between OTO and HR;
2. Opposite Tibia Vertical (OTV) between HR and OIC;
3. Opposite Heel Rise (OHR) between FA and TV.

This addition creates symmetrical 10-phase gait cycle that now can be easily divided in two parts (phases in bold are new):

1. IC → OTO → **FA2** → HR → **OTV** →

2. OIC → TO → FA → **OHR** → TV.

Fig. 4.2 is an extended version of Fig. 4.1 and shows all 10 phases and the positions of considered gait events during gait cycle.

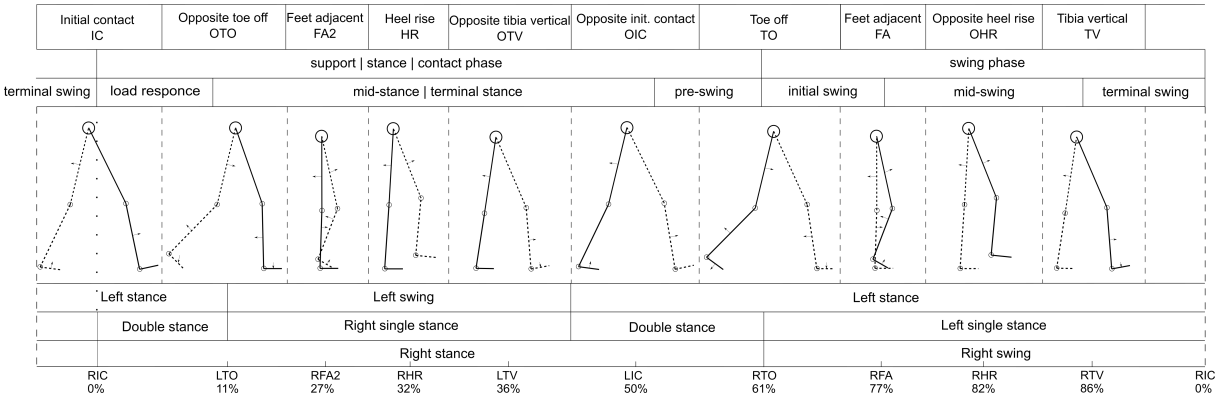


Figure 4.2: Detailed 10-phases gait cycle.

The considered extended gait cycle allows to identify simulated and measured gait's characteristics. Starting from gait's overall period and initial phase; through characteristics of each gait's phase; to quantifiable variation in gait cycles.

4.2 Experimental data

One of the distinctive qualities of gait simulations is the availability of all required characteristics. Such kinematic values as positions, velocities, and accelerations can be measured externally from a real body motion. But exact kinetic (dynamic) values of force and torque as well as neuro-muscular values of excitation and muscle activation are hard to measure and evaluate. Neuro-musculoskeletal models allow to calculate any required value, taking into account precision and limitations of the model.

In this thesis, for model comparison and evaluation, we use kinematic values of limbs and activation of muscles that are close to skin and can be measured with electromyography (EMG). Initially, for comparison procedure prototype with GAIT2DE model, we have been using averaged curves of joint angles and some muscle activation of healthy human gait from `https://musculoskeletalkey.com/normal-gait/`, see figures 2.10 and 2.5 on site. They denote a single walk by a 22-year-old normal female, weight 55 kg, walking barefoot with a cycle time of 0.88 s, a stride length of 1.5 m and a speed of 1.70 m/s. The plots were manually traced and digitized into MATLAB. This resulted in a plot on Fig. 4.3.

Fig. 4.3 shows muscles' names in terms of GAIT2DE model. Due to limitations in EMG measurements, electric activation signal of a muscle affect nearby ones. Quadriceps muscle combines Vasti group and Rectus Femoris muscles. Triceps Surae combines Soleus and Gastroc muscles. To compare these average activations with signals from individual muscles in the model, combined signals were separated using linear windows. The windows are two opposite piece-wise functions for each muscle, which transit between 0

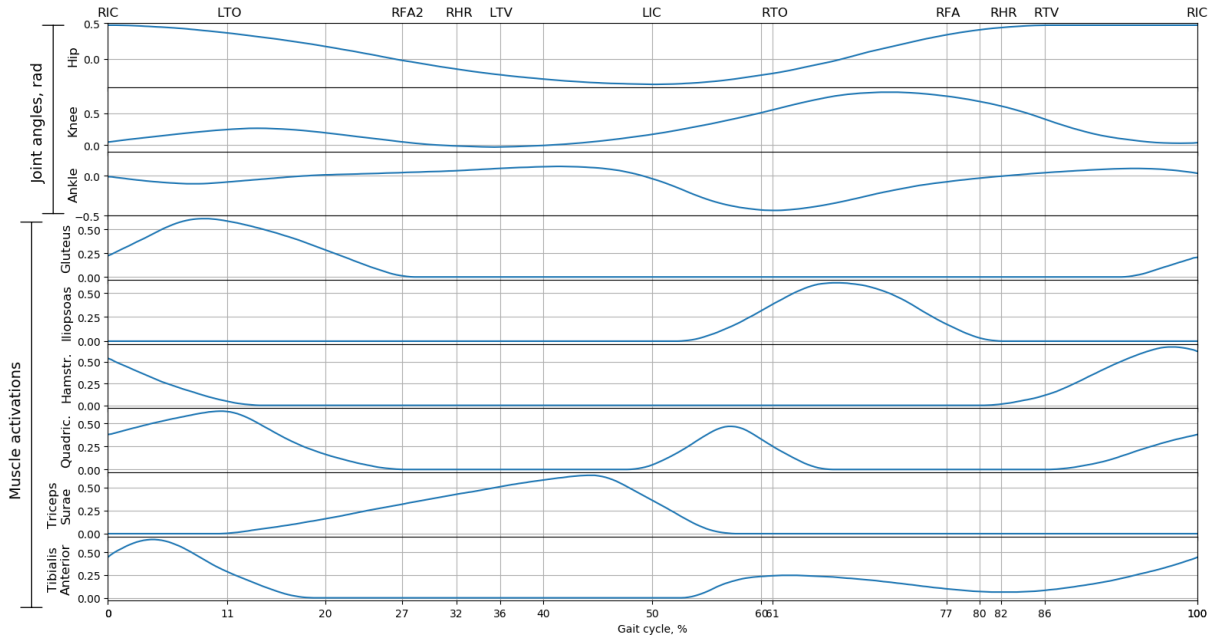


Figure 4.3: Initially used experimental data with gait cycle marks as on Fig. 4.2. Extracted from `https://musculoskeletalkey.com/normal-gait/`.

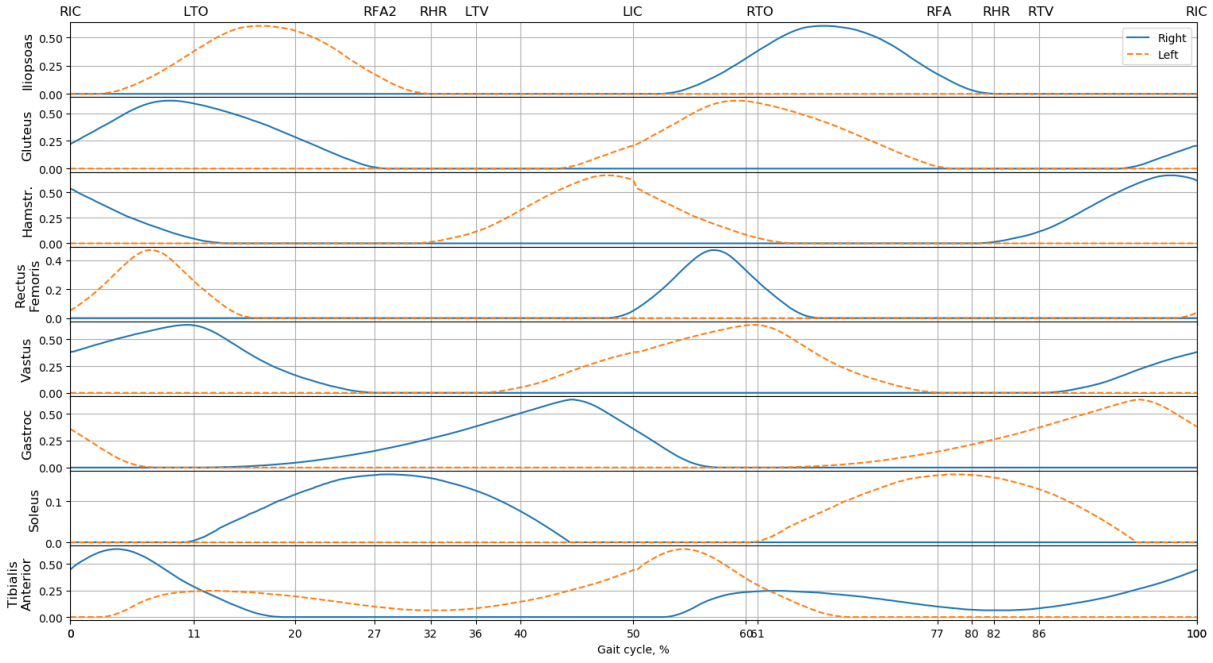


Figure 4.4: Muscle activation from Fig. 4.3 with separated combined muscle EMG for both legs.

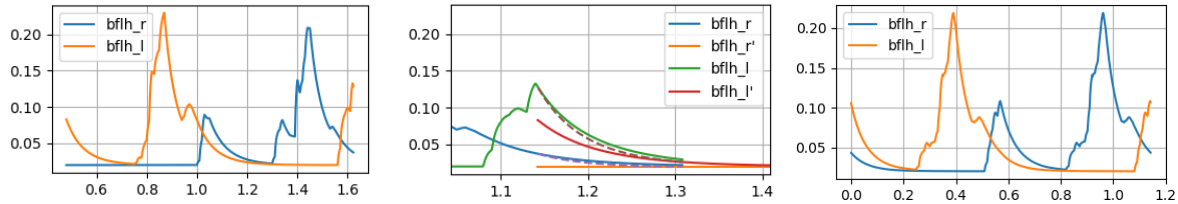
and 1 linearly in the activation region to separate. Borders of separation windows were defined empirically from anatomical function of muscles. Separated muscle signals for both legs are shown on Fig. 4.4.

Simulated muscle data from OpenSim model

Previous muscle data source was adequate to use with GAIT2DE model (see Section 3.1), but OpenSim model has up to 40 muscles per leg and allows to use much wider set of neuro-muscular values (see Section 3.2). The authors of said model have performed simulation tests to evaluate the correctness of the model [4]. These tests include inverse-dynamic calculation of muscle activation from motion-captured human walk and a further comparison of simulated muscle activation with measured subject's EMG of several muscles. The 31-year-old male subject (height 182 cm, mass 85 kg) has performed a single gait cycle at self-selected speed.

The correctness test results was good, taking into account small difference between simulation and EMG and a little input of torque drives into joints' movement. These factors make the usage of model-originating gait simulation data convenient for optimizing CPG controller parameters and analyzing results.

Fig. 4.5a shows extracted simulation data for an example muscle, Biceps Femoris Long Head of right and left leg (bflh_r, bflh_l). But prior to optimization, simulation data of walk segment needs to be transformed into periodic gait cycle. It requires the equality of length and time steps of compared signals. The extracted values hold only one period of normal gait, therefore require preprocessing before being used. This includes equalizing the start and final point, accounting for time resolution and starting phase. Transforming comprises of four steps, each dealing with one issue of observed human gait segment.



(a) Example initial extracted muscle activation of both legs.

(b) Difference in muscle activation at the start and end of period.

(c) Final muscle activation, fit for further comparison.

Figure 4.5: Transforming experimental data for simulated gait analysis.

1. The original time grid is uneven (has varying Δt). Using rounded median time step of $\Delta t = 0.7$ ms, new curves were obtained by interpolating from original uneven time grid. The new Δt is also much smaller than is used for gait simulations ($\Delta t = 5$ ms).

2. The walk segment is longer than a single gait cycle. Measured gait is a walk segment of about 1.6 seconds. From knee angles, a gait period of $T = 1.1424$ seconds was empirically chosen. The extra time was used in the next step.

3. Observed gait is not periodic. This means that the last point of a signal period is not equal to the first one. It was fixed by applying a linear window over the extra part (from previous step) and the

equal time over the signal's start (Fig. 4.5b). The dash on the figure (bflh_r') denotes the starting part of the same signal drawn over the ending part.

4. Observed gait is not symmetric between legs. To deal with that, an average curve with respect to half-phase shift of second leg has been calculated and resulted in final curves for both legs on Fig. 4.5c.

All mentioned steps are aimed to take advantage of standard human gait cycle. The gait cycle allows to recognize walking events, divide gait period onto phases, and analyze the resulting gait phase-by-phase.

4.3 Optimization of controller parameters

In this work, the musculoskeletal model is considered unchangeable (its parameters are not optimized but ought to fit human subject), but the values of CPG parameters and weights of connections are in question.

There is not enough insight about functioning of CPG network, left alone our implementation of neural circuitry. Because of that, the correct way of CPG function is unknown, so the parameters of controller must be evaluated from the overall quality of gait after simulation of several steps.

Gait controller contains from 44 to about 130 parameters, depending on composition of the controller. Parameters for right and left leg are the same. Considered amount of CPGs in controller was 4, 6, or 8 CPGs, whose architectures are described further in Sections 5.4, 5.2, and 5.3 respectively.

Each CPG has 2 half-centers with 4 layers (see Section 2.3). Half-center parameters are the same, except weights from Rhythm Generator (RG) to Pattern Formation (PF) layers, W_{PF} and W_{PE} in flexion and extension half-centers respectively. Also, weights from Motoneuron (MN) layer to muscles are individually tuned.

RG parameters are defined manually from several simulations and the work of Jouaiti et al. [72]. Weights from RG to PF are always $W_{RG} = 1$.

Finally, each sigmoid neuron has two internal parameters and several input weights. Each of 2 to 4 types of CPG contain 2 sigmoid layers, up to 3 proprioceptors and optional exteroceptors of trunk angle and ground force.

4.3.1 Parameter optimization methods

Current implementation of forward dynamics simulation is single-threaded. With GAIT2DE model, the calculations are about two times faster than real time, but OpenSim's model has simulation speed of up to one minute of calculations per second of gait on 4 GHz processor thread. Although, any amount of simulations could be arranged in parallel.

This leaves us with the necessity to optimize an expensive target function, when each set of parameters is evaluated for up to ten minutes of calculations.

Up to that, parameter space of "normal" gait has a lot of local minimums, regardless of target functions, simply due to degrees of freedom problem [49].

BlackBox

For OpenSim model, the usual optimization methods like genetic optimization were considered to be unsuitable due to high amount of expensive function calls within each iteration. Thus, we have found and used BlackBox, a Python module for parallel optimization of expensive black-box functions [106] (`https://github.com/paulknysh/blackbox`).

The inputs for BlackBox are: a target function, which is an average correlation to human kinetic and biological values, described further; a search domain, which is a list of ranges for each CPG parameter; and two numbers of allowed target function calls (n, m), which define search accuracy vs. optimization time.

BlackBox works in two stages, each calling the target function the allowed number of times. In stage one, it calculates function in n random, evenly placed points in d-dimensional space. From those points, an approximated multidimensional surface is reconstructed, where the minimum is searched in the next stage. For that, another m points are calculated with mCORS (modified Constrained Optimization using Response Surfaces) algorithm to find the final minimum. This approach allows to reduce expensive function calls but requires the parameter hyper plane to be relatively smooth.

46 CPG parameters were optimized with BlackBox. Parameters were connection weights, α's and θ's (initially randomized within empirically-chosen limits). Thus, BlackBox parameters were $d = 46$, $n = 2 * d = 92$, $m = 2 * d = 92$. Objective functions are described next.

4.3.2 Target functions

Describing an overall quality of gait as a single value is extremely hard, even when this gait is simulated and all gait characteristics are available. Through the development process, I've been using three incrementally complex functions.

4.3.2.1 Distance of walk

The easiest way to evaluate gait is a distance of successful walk. Successful walk is defined to be without falls or within some limits of some parameter (e.g. maximum trunk angle). Walk distance value has been used during initial search of CPG parameters.

Later, having a decent controller, the distance of successful walk during a defined time (e.g. 10 seconds) may give a mean velocity of gait. Bigger velocity of gait while tracking vital constraints of gait, tells about successful cyclic changes of gait phases and absence of stumbling upon the ground.

4.3.2.2 Energy cost of gait

Velocity of gait is still not a very good value for optimization as in pursue for a faster walk, optimization method eventually ends up with unrealistic or impossible gait. While the latter can be omitted with stiffer constraints, realistic gait has rather wide borders. Pachi & Li [91] have reported limits of realistic walking period from 0.8 to 2 seconds.

Naturally, among all control strategies for gait, the optimal ones are chosen to reduce energy consumption of body. There are several values of energy consumption available for optimization, such as energy per time, distance, or cost of whole action [107].

In this thesis, a simple mechanical work of each muscle has been used, in addition to distance. Each time frame and for each muscle it is calculated as $W = max(0, F \cdot \Delta x)$, where F is force of a muscle, Δx is change of its length between frames. Only positive performed work is accounted.

The proposed optimization value is $optimval = -100 \cdot D + W$, where D is maximized traveled distance, which is set to the same order as work W to be of the same value for optimization. Total gait duration was set to 10 seconds, so $D = 1 \dots 10$ meters and $W = 100 \dots 1000$ Joules.

The energy cost of gait is useful during final tuning of CPG parameters for smooth and efficient gait, but even the approximate values of CPG parameters are unknown, so we would need an intermediate target value to describe natural gait.

4.3.2.3 Gait parameters evaluation

Finally, neither distance, nor work alone allow to quantify realistic gait. The best approach is to evaluate kinematic and kinetic gait characteristics, but that is currently not possible. Chapter 1.1 tells that the complete kinematic and kinetic gait classification is not available. And the creation of such gait classification routine goes beyond this work.

Thus, a direct approach has been chosen, to compare kinematic and kinetic parameters of simulated gait with a measured ones from human. We do not perform any experiments or measurements, the data from open sources has been used. It is described in the next Section 4.2.

The comparison procedure was implemented for purposes of controller optimization and a very simple classification of simulated gait. The optimization target function holds the idea to evaluate the overall quality of gait by comparing dynamic values of simulation to corresponding human values during gait. The classifier describes gait simulation with comparison values and additionally with values of gait period and initial phase.

So, the aims of considered comparison-based gait evaluation criterion would be:

- Compose an optimization function regarding quality of gait using kinematic and kinetic values.
- Compare simulated gait with measured data from healthy people.
- Use the modified procedure with non-healthy gait as a part of gait type classifier.

4.4 Correlation gait evaluation criterion

In this section, prior described experimental data (see Chapter 4.2) is compared to simulated gait to quantify the similarities. Generally speaking, the procedure is as follows. A correlation matrix is calculated for each gait value of joint angles and muscle activation. The matrix is composed of correlation values for each gait period and phase. The resulting matrices are then combined to calculate an average correlation matrix. The maximum value in this matrix is the similarity value between simulation and the human gait. And the position of the maximum tells about the closest initial phase and frequency of gait.

Then, with the help of described gait cycle (see Chapter 4.1) and initial phase and frequency of gait values, the simulated gait can be divided onto phases and estimated further in detail.

4.4.1 Cross-correlation

The basic comparison procedure is calculation of a central point of cross-correlation function (scalar product or dot vector product). It was chosen for its simplicity and suitability and has been used in author's precedent works [108].

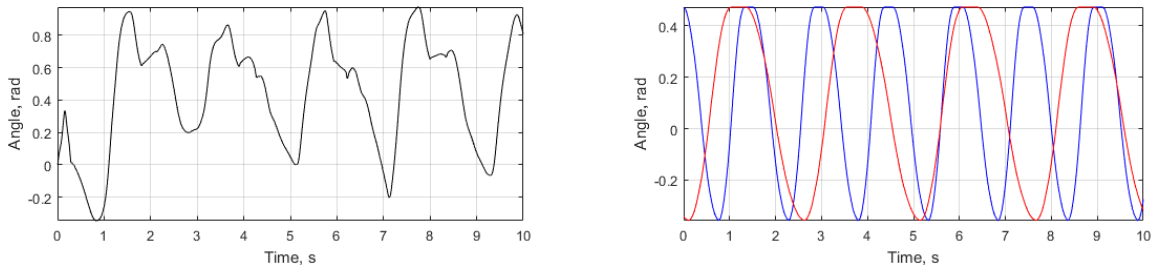
Cross-correlation of two normalized vectors shows sample-wise similarity of their shapes. Geometrically, this operation can be described as length of projection of one multidimensional vector onto another. Normalization sets length of a vector to 1 and "shape" means mutual ratio of samples of a normalized vector.

This operation is sensitive to shifts in signal's initial phase and difference in frequency. Because of that, before deciding on gait quality, one must know the exact frequency and phase of simulated gait. Luckily, the same sensitivity property allows to find out the phase and frequency by calculating the reaction to each pattern.

The criterion application is illustrated with joint angles of one cycle from Fig. 4.3 with modeled gait curve of 10 seconds with GAIT2DE model.

First, a series of gait patterns is built, with periods from 1.5 to 2.5 seconds (expected gait cycle duration). Then, each pattern has its phase shifted along the whole period. Building a pattern for each period and shifted phase for each time step may be computationally expensive. By making shift step to every 10 time steps, the accuracy of results is reduced only a little. Fig. 4.6b shows two example patterns with the smallest (blue) and the biggest (red) used periods. Red curve is also shifted for half of period.

Then, each of the patterns is compared via cross-correlation with corresponding gait curve (right hip joint angle on Fig. 4.6a) resulting in a matrix of scalar products.



(a) Angle of right hip from simulated gait.

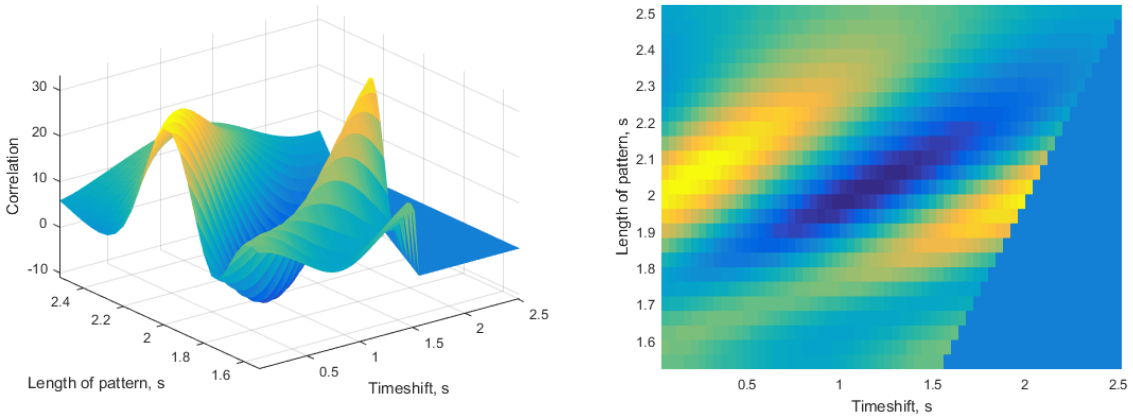
(b) Example patterns from experimental signal.

Figure 4.6: Gait signal generated by our CPG model and examples of constructed patterns designed to match gait signals.

A matrix of such scalar products with different periods and shifts form some sort of a spectrogram-like surface, but unlike Fourier and wavelet spectrum, these patterns aren't mutually orthogonal (independent). As we don't do decomposition but compare curves and calculate similarities, the lack of orthogonality doesn't neglect the result.

Corresponding surface is shown on Fig. 4.7, side and top views. The rows of this matrix are periodic (right part just repeats itself) so we filled repeated part with zeroes. And the range of correlation values is transformed from [-1;1] to [0;1]. Two peaks on the plot are actually the same, repeated, peak. This peak corresponds to the pattern that is the most similar to the simulated gait. And the coordinates of

this peak, i.e. period and initial phase, describe the simulated gait.



(a) Side view (b) Top view

Figure 4.7: Matrix of correlation values. Timeshift is synonymous to initial phase and length of pattern is the period.

The maximum value of the matrix corresponds to pattern on Fig. 4.8 (blue periodic line) drawn over simulated angle curve (black line). The pattern has period of 2.04 s and time shift of 0.24 s.

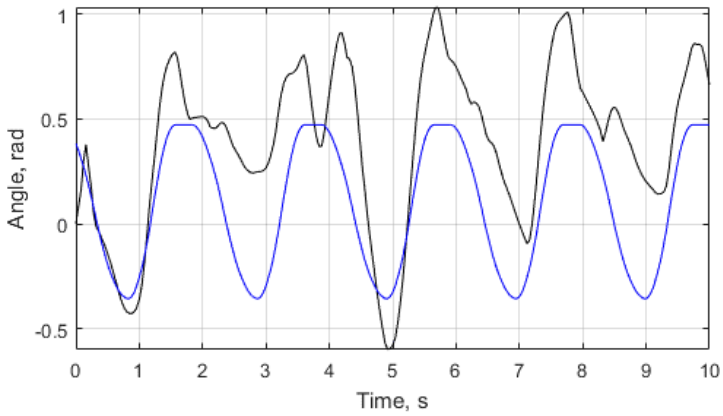


Figure 4.8: Pattern with biggest correlation value over model angles.

Next, the correlation matrices are calculated from the other experimental data. Fig. 4.9 shows matrices and best patterns for all leg joints. Periods are close, from 2.04 to 2.12 s, but phases are all different: 0.24, 0.16, 1.36, 1.4, 1.08, and 0.44 seconds. There should be a single delay between right and left leg (about a half-period, 1 second). From this example, ankle clearly stands out due to its more complex anatomy and control.

To determine a single period and delay value for all experimental data, we simply calculate an average of all considered correlation matrices and look for maximum value.

Finally, the target function returns a correlation value in range [0;1], indicating the similarity of resulting gait to human walk and corresponding period and delay values that describes gait and allow to

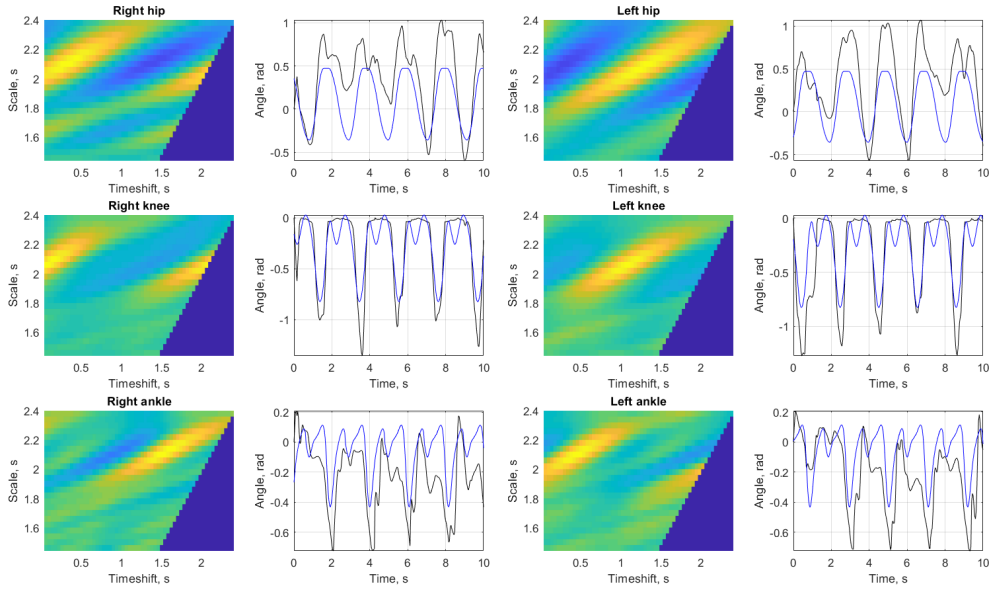


Figure 4.9: Matrices of correlation values (top view) and fitting patterns (in blue) for all leg joints.

divide simulated gait onto phases for further analysis.

4.5 Conclusion

This chapter has described the means of simulated gait analysis, used in this thesis work and, based on the former, the controller parameters optimization.

The periodic gait cycle was described to divide gait into easy to recognize events. The initial gait cycle was augmented to provide a more dense and symmetric cycle. The human data has been described then and the ways to bring it on par with the simulated gait. The human data, composed of leg joint angles and muscle activation, have come from two sources: generalized human gait cycle and from the measurements and simulations of the OpenSim musculoskeletal model authors.

Simulated gait evaluation and controller parameter search was performed by comparing simulation to real gait using correlation value. For the optimization, BlackBox algorithm has been used. Two other optimization values have been described, suited for initial and final CPG controller parameter search.

The next chapter provides simulations of human gait with CPG controller in different architectures.

Chapter 5

Results of neuro-musculoskeletal simulations

In order to illustrate the ideas of CPG control of human gait, we have performed computational simulations using aforementioned controllers and musculoskeletal models. Each section of this chapter describes an architecture of neuro-musculoskeletal platform as connections between its main parts followed by the simulation results themselves.

But first, a description of the connection of CPG controller to the musculoskeletal model is in order with an example simulation of a single joint control. Then, the results of gait simulations using different composition of CPG network in the controller. They differ in the main principle of control, targeting a joint, an antagonistic pair of muscles, or a gait phase.

To remind the reader, closed-loop CPG control of gait is performed as follows:

1. Brain controller initiates movement, suppress reflexes, and alternates velocity and direction;
2. Spinal controller excites muscles through motoneurons;
3. Musculoskeletal system performs movement and returns sensory feedback to both controllers.

5.1 CPG control of a single joint

The main control principles are illustrated by simulation of a single joint controlled by two muscles of GAIT2DE model. This simulation example was presented at UKRCON17 conference [102].

The joint to be controlled is the human hip joint, actuated by Iliopsoas and Gluteus muscles (Fig. 5.1a). It is the primary joint involved in the human walk. The two muscles are only involved in hip movements, their parameters are given in Table 5.1. Iliopsoas is the hip flexion muscle that turns thigh forward, Gluteus is actually a group of three hip extension muscles, that are used as one here that turn thigh backwards.

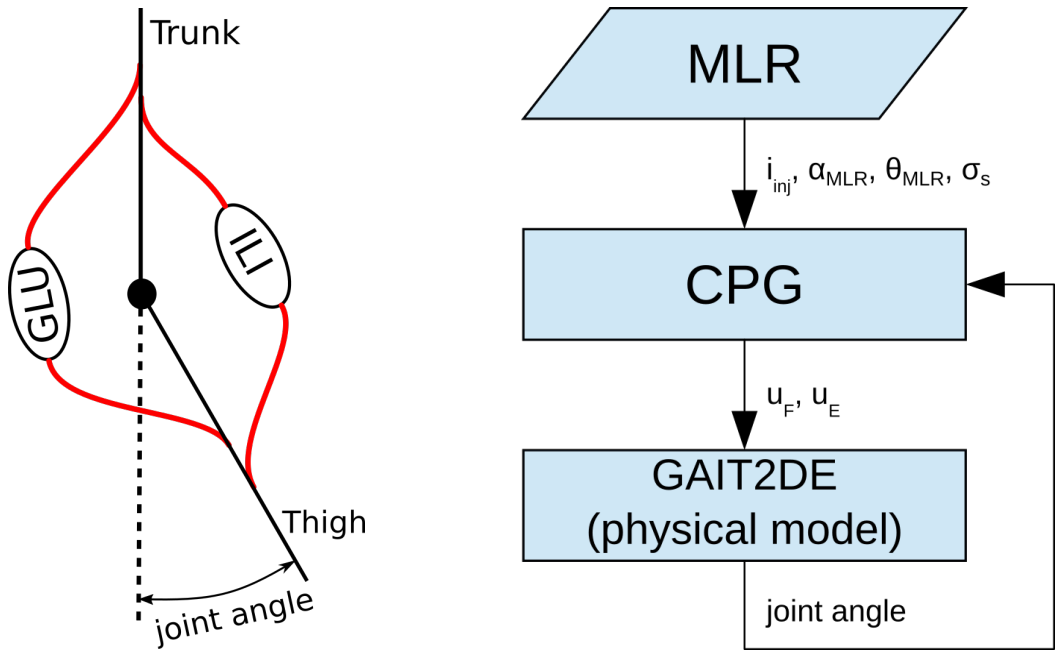
Since this simulation involves control of only one joint, the model must kept in neutral conditions. Meaning that the model should not fall, interact with ground, or be affected with its swinging mass. This was made by resetting the model's spatial coordinates at every time step, resulting in infinite fall. To

Table 5.1: Parameters of hip-only muscles of GAIT2DE model (extracted from [97]).

Muscle	Muscle mechanics				Muscle-skeleton coupling	
	$F_{max}(N)$	$L_{CEopt}(m)$	$\frac{L_{slackPEE}}{L_{CEopt}}$	$L_{slackSEE}(m)$	$L_0(m)$	$d_{hip}(m)$
Iliopsoas	1500	0.102	1.200	0.142	0.248	0.050
Gluteus	3000	0.200	1.200	0.157	0.271	-0.062

simulate the neuro-musculoskeletal system of the human hip, the GAIT2DE simulator was modified (see Section 3.1) in order to send the signals generated by the CPG to the muscle model.

The full scheme of simulator (Fig. 5.1b) consists of brain controller "MLR" that projects to CPG composed as on Fig. 2.2b, except that joint angle is used as feedback to the controller. CPG controller forms muscle excitations u_F and u_E to two muscles that move hip joint. Signals from MLR are sinusoidal i_{inj}, σ_s, and arbitrary α_{MLR} and θ_{MLR}. i_{inj} enables initial RG de-synchronization and could provide its rate for σ_s to adopt [82]. σ_s influences RG frequency as a cell parameter; α_{MLR} and θ_{MLR} control PF neurons' conductivity, their coupling to MN and balance between flexion and extension. In this simulation, $\alpha_{MLR} = 1$ and $\theta_{MLR} = 0$ to disable clamping. More on CPG model in Chapter 2.



(a) Right hip joint of the GAIT2DE model.

(b) General scheme of control.

Figure 5.1: One-joint neuro-musculoskeletal simulation.

RG equations used here are defined in Eq. 2.7 with constant σ_s. Constants' values are $\varepsilon = 0.1$, $\tau_m = 0.5$, $\tau_s = 10.0$, $A_f = 1.0$, $\sigma_f = 30.0$, $W_o = 0.1$. With these parameters, the resulting RG frequency in the range of $[0.1; 2]$ Hz can be approximated with Eq. 5.1.

$$\sigma_s(f) = 198.5 \cdot f^2 + 3.5 \tag{5.1}$$

where f is desired frequency of RG oscillation in Hz.

Two simulations with single joint are presented. One 10-second simulation of constant swing frequency of 0.6 Hz which according to Eq. 5.1 corresponds to $\sigma_s = 75$. Another 60-second one, where the frequency is changed halfway through the run from 0.6 Hz to 0.3 Hz ($\sigma_s = 21$).

5.1.1 Movement with constant speed

In the first simulation, RG half-centers provide oscillations of arbitrary amplitude and defined frequency to PF neurons (Fig. 5.2). RG layer here has no input from MLR. Note the specific shape of RG patterns — almost sine, but with the rising edge steeper than the falling edge.

PF neurons transform input value from RG to range $[0; 1]$ (Fig. 5.3a). Flexion/extension domination and rhythm deletion through changing α_{MLR} and θ_{MLR} in PF layer aren't applied. PF neurons are connected to MN layer, whose parameters are set to make joint movement symmetrical and softly limited to ± 0.5 rad by SN.

MN layer produce muscle excitation: u_F for flexor Iliopsoas and u_E for extensor Gluteus, along with initial state of the model and optional external forces and moments applied to body parts and joints. Flexor excitation peaks (connected to Iliopsoas) are bigger than those of extensor because Iliopsoas muscle is weaker than Gluteus (Table 5.1). To close the control loop, CPG contains SN for articular reflex [82] that reacts to angle of joint to produce inhibitory influence on each motoneuron to prevent their turning above desired values. The last two plots on Fig. 5.3a show the length of CE element of muscles and the force exerted by the muscles.

Fig. 5.3b shows phase diagram of simulated movement of the joint. It starts from resting position and in one period establishes at a stable trajectory. Abrupt changes of trajectory near ± 0.5 rad show articular limitation from sensory neurons.

5.1.2 Movement with variable speed

In the second simulation, the frequency of RG neurons is changed halfway to show the movement mode transition, i.e. slowing down from running to walking. Fig. 5.4 shows the same plots as Fig. 5.2 and Fig. 5.3a with the change of σ_s value. Prior transition, the movement is stabilized, then at $t = 30$ s σ_s is changed to reduce the oscillation frequency from 0.6 Hz to 0.3 Hz. The change of σ_s is instantaneous, but the model adapts to new frequency during the next period in a smooth, but rapid transition. Joint angle plot shows a single larger forward swing after the transition due to sparser extension MN pulses and non-adapted SN.

The force plot on Fig. 5.3a shows that for a less frequent movement not only the muscle contractions are sparser, but also have lower amplitude. This reduction is partially explained by less amplitude of muscle excitation from MN layer.

The same transition can be seen on the phase diagram of Fig. 5.5. Initial trajectory is equal to previous setup but then the joint produces one swing with higher amplitude and stabilizes on second trajectory.

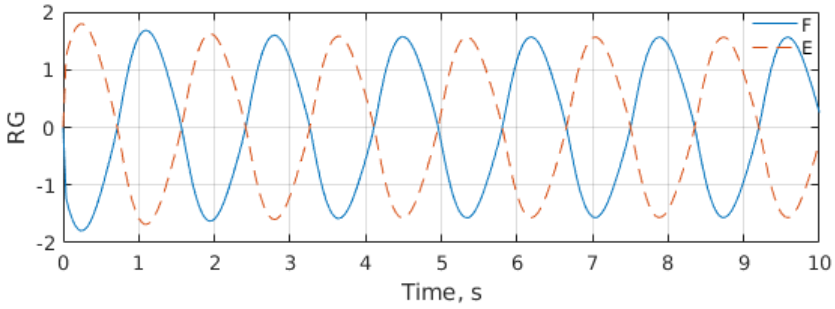
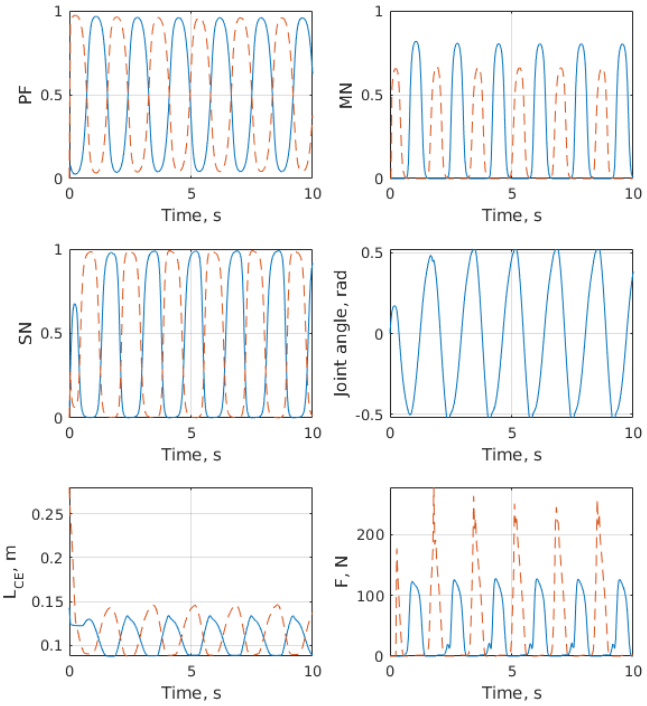
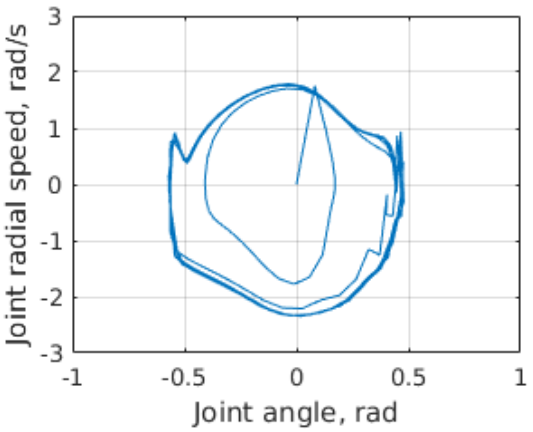


Figure 5.2: Rhythm generator with no input. F for flexion RG half-center, E for extension.



(a) Signals from CPG and physical model. Blue solid line is flexion half-center, red dashed is from extension half-center.



(b) Phase diagram of constant hip joint movement.

Figure 5.3: Constant frequency joint control.

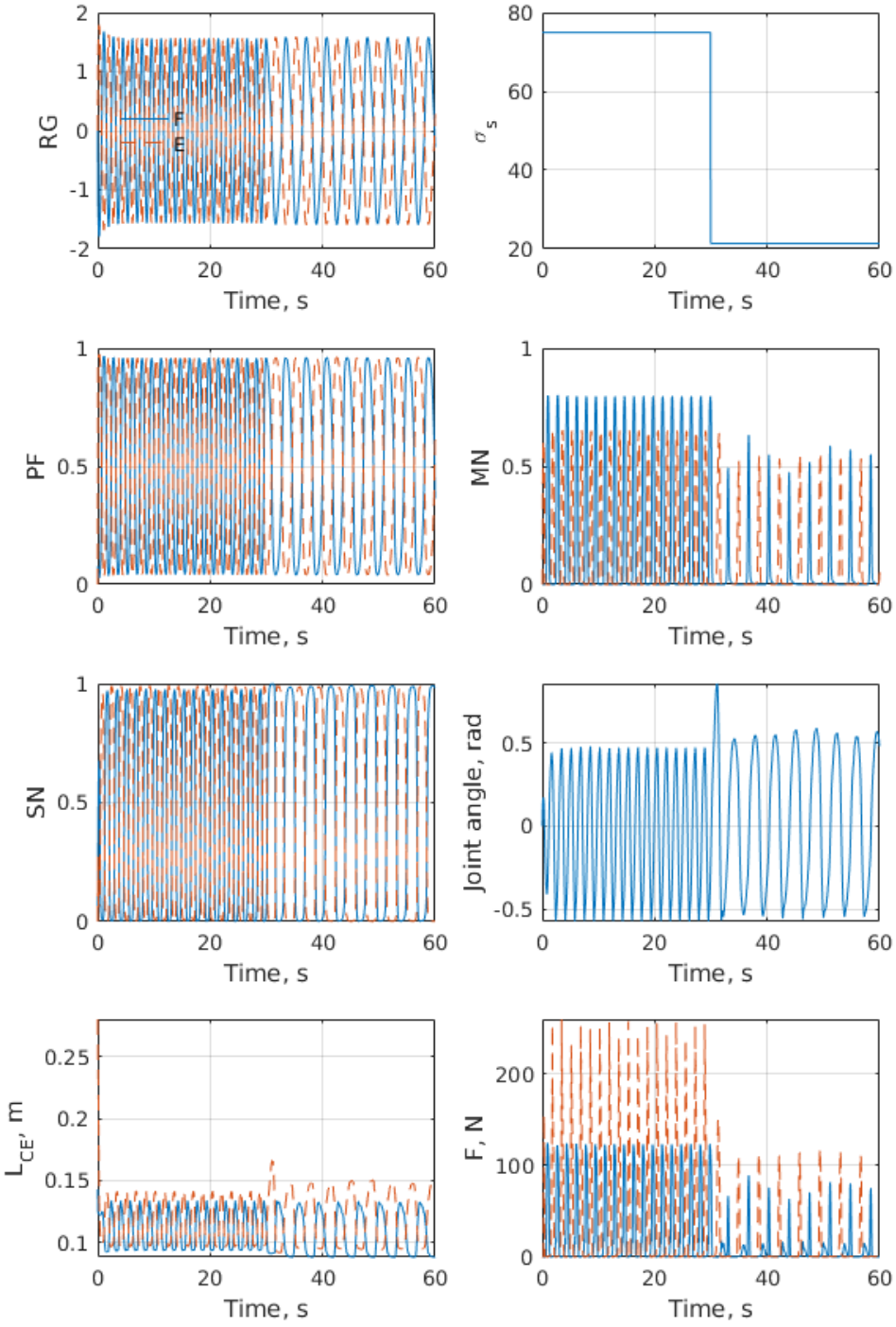


Figure 5.4: Signals from simulation with RG frequency change. Blue solid line is flexion half-center, red dashed is extension half-center.

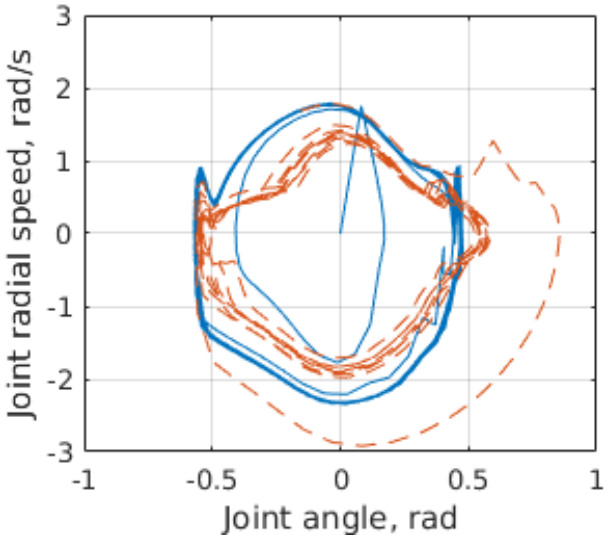


Figure 5.5: Phase diagram of varied hip joint movement. Solid blue line is before frequency change, dashed orange is after the change.

Thus, the simulator shows the ability to generate rhythmic movements for controlling a biological joint and variate their parameters which can originate from upper controller. Next section shows an implementation of several CPGs to control lower human body and simulate gait.

5.2 Joint-targeted CPG controller

A joint-targeted CPG network architecture is based on an idea to dedicate a CPG for each joint. For human gait control, a network of six interconnected CPGs is required. CPG of right and left leg are symmetric, with opposite oscillation phases. CPG for hip, knee, and ankle joints differ in sensory input, parameter values, and amount of controlled muscles.

This approach works well with small amount of monoarticular muscles, e.g. two muscles per joint, that affect only their own joint. The addition of biarticular muscles has shown the lack of CPG units to control them. In further cases, the other CPG architectures should be considered such as a CPG per antagonistic pair/group of muscles or a CPG for each phase of movement.

In this simulation, all joints of GAIT2DE models were used, along with twelve out of sixteen available muscles. The excluded muscles are Hamstrings and Rectus Femoris biarticular hip muscles. They are denoted as HAM and RF on Fig. 3.2b. As described before, they are not consonant with the controller target and greatly complicate the gait control. It is acknowledged that such exclusion goes against this work's aims to be as biologically plausible as possible, but this control approach is simple and appropriate for initial setup in gait control.

This control approach and simulations were presented at BioCAS17 conference [90].

Fig. 5.6 shows network of 6 CPGs which was used with GAIT2DE simulator for musculoskeletal system. As in previous section, CPG for each joint arranges and passes output of its MN to musculoskeletal system of GAIT2DE simulator. The latter then passes the feedback to CPG network. One CPG indeed

controls an antagonistic pair of muscles, and hip CPGs are interconnected together to coordinate the limbs. Force sensors are implemented on soles to return pressure force on heels and toes to distinguish gait phases. A virtual elastic attached to the top of trunk and able to slide horizontally (dotted line on Fig. 3.2b, 5.14) as lifting support harness.

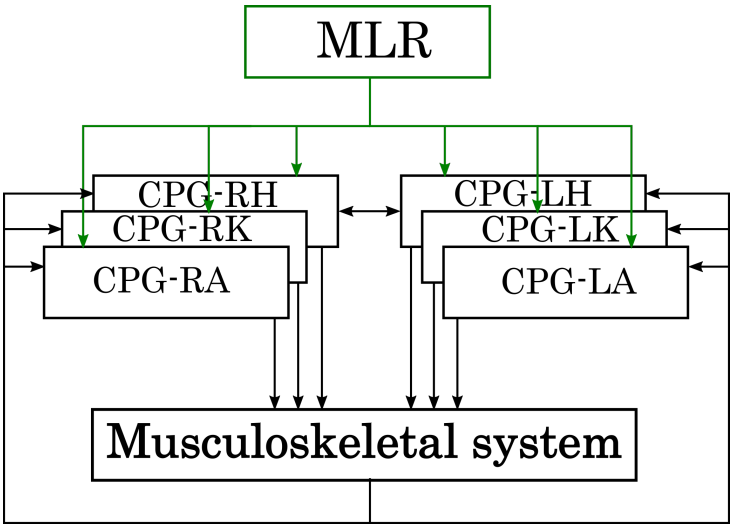


Figure 5.6: Simulation setup of 6 CPG with interconnections. CPG names stand for Right and Left Hip, Knee, and Ankle respectively.

All joints follow the same connection scheme as on Fig. 2.2, except that knee CPG's PF neurons have additional connection (excitatory for flexor and inhibitory for extensor) from hip's flexor SN that corrects knee phase; and ankle CPG's PF neurons have two ground sensory neurons (SNG). SNG of toe ground reaction is connected to flexor PF, heel SNG is connected to extensor PF. Fall SN measure trunk angle and innervate ankle CPG's PF neurons to help preserve equilibrium.

The CPG parameters were chosen manually according to each parameter's purpose. This was possible due to simplicity of setup, including CPG controller and musculoskeletal model. Each CPG controls its own pair of muscles and has its own role according to the functioning of a joint.

5.2.1 Walking with constant speed

In a simulation of 10 seconds of gait with constant speed, the model is able to walk for about 5 meters with 10 steps. Fig. 5.7 shows seven consecutive frames of gait simulation. The model was tested to walk for at least a 100 seconds. Fig. 5.8 show CPG activity, muscle excitations, and joint angles for hip, knee, and ankle respectively. Walking simulation starts from double support phase. Transient phase can be observed at the beginning of the walk on the muscle activities. Supporting elastic slightly contributes to oscillations of trunk like a real harness.

Resulting joint angles are qualitatively similar to human walking cycle [104, 110], especially hip and knee joints that are similar to those on Fig. 5.8d, where articular angles of the human hip, knee, and ankle are compared to those with a rigid and non-rigid human foot soles. Indeed, this simulation corresponds to the case of rigid soles as knee and ankle joint angles are closer to rigid plots on Fig. 5.8d.

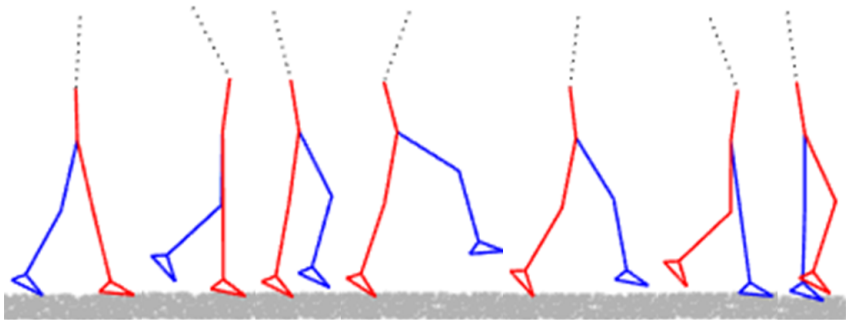
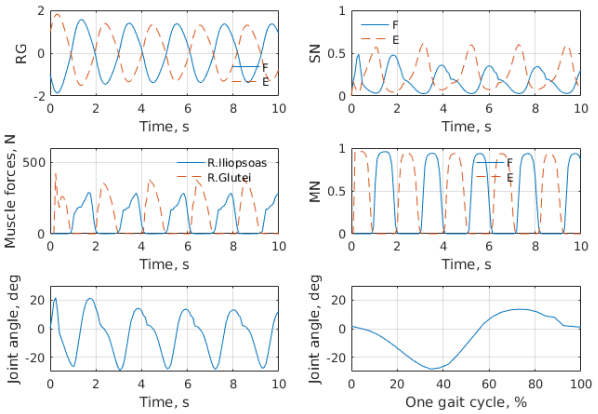
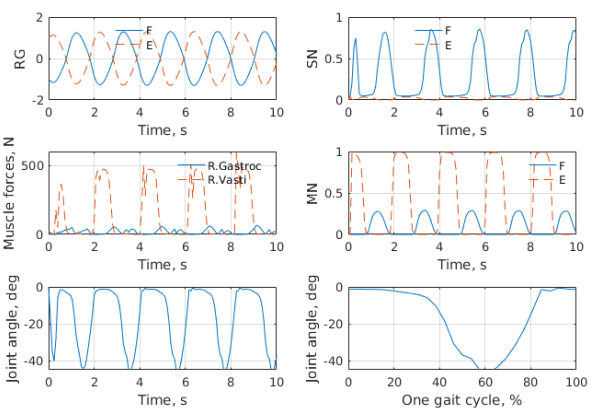


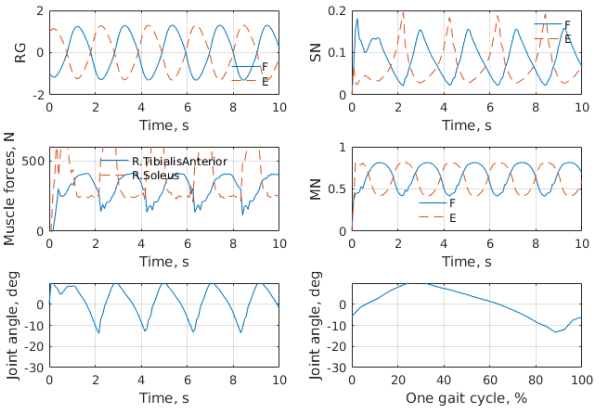
Figure 5.7: Several consecutive frames of GAIT2DE model simulation, time between frames is 0.1 s.



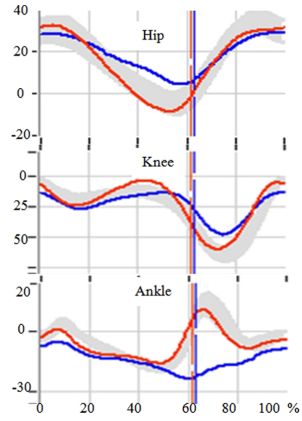
(a) Right hip CPG.



(b) Right knee CPG.



(c) Right ankle CPG.



(d) Articular angles (in degrees) of the hip, knee, and ankle, with a rigid (blue line) and non-rigid (red line) foot soles during one gait cycle. Extracted from [109].

Figure 5.8: Walk simulation with constant speed. Signals of right leg. F and E denote flexion and extension RG half-center, sensory neuron, and motoneuron respectively.

5.2.2 Walking with variable speed

Frequency of RG is controlled by neural parameter σ_s which acts like a frequency gain in a rhythmic activity. Changing σ_s online is similar to reproducing neural plasticity effects [72]. Indeed, σ_s can be modified by a learning or adaptive law modeling a homeostatic effect. But in this simulation, σ_s was changed directly and instantaneously. σ_s should always be a positive value, because of a square root in learning rule (see Eq. 2.7).

After σ_s has changed, RG need some time to stabilize (as on Fig. 5.4, 5.5). CPG produces one swing with higher amplitude and then stabilizes on second trajectory. σ_s shouldn't be changed instantaneously in applied control systems and is made in sake of studying the CPG controller.

Fig. 5.9 shows variable sagittal velocity of pelvis of biped (Fig. 3.2b) and its moving average (sliding window width is 8 seconds) reacting to manual change of σ_s as square signal. After change of σ_s parameter, model needs about 3 seconds to stabilize its speed showing the global stability of the gait.

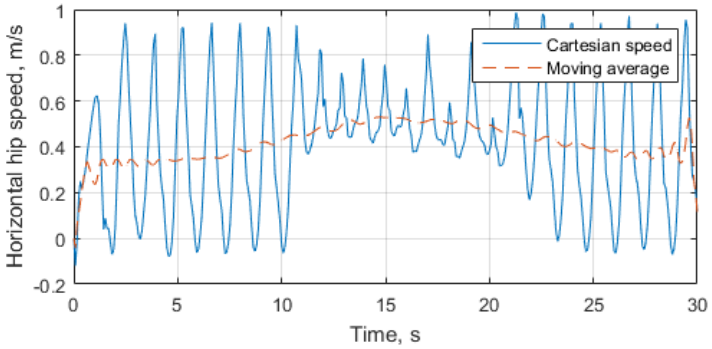


Figure 5.9: Horizontal hip speed and its moving average. σ_s starts at 30, changes to 50 at 10 seconds, and back to 30 at 20 s.

5.3 Muscle-targeted CPG controller

A muscle-targeted CPG architecture holds an idea to dedicate a CPG to solely control two antagonistic functional groups of muscles. This idea extends the previous architecture in case of growing number of muscles in the model. Now, the Hamstrings and Rectus Femoris muscles are not excluded, but establish a new group of muscles. This group includes muscles that affect both hip and knee joints. Many more muscles are added to these four groups based on their main gait function.

This section describes gait control using a controller with 8 CPGs, connected in series (Fig. 5.10). CPGs are symmetrical between legs, but each CPG of a leg is slightly different in its connections and feedback. All CPG contain the same 4 levels: Rhythm Generation, Pattern Formation, Motoneuron, and proprioceptive sensors (see Chapter 2).

Fig. 5.11 shows the details of each of four CPG types. The controller receives the same descending signals from spinal cord, i_{inj} for gait mode switch, α_{MLR} for changing stride length, θ_{MLR} for start/stop of gait. And now we are adding a new signal, $sens\_sens$, for selective sensory reflex inhibition during different gait modes.

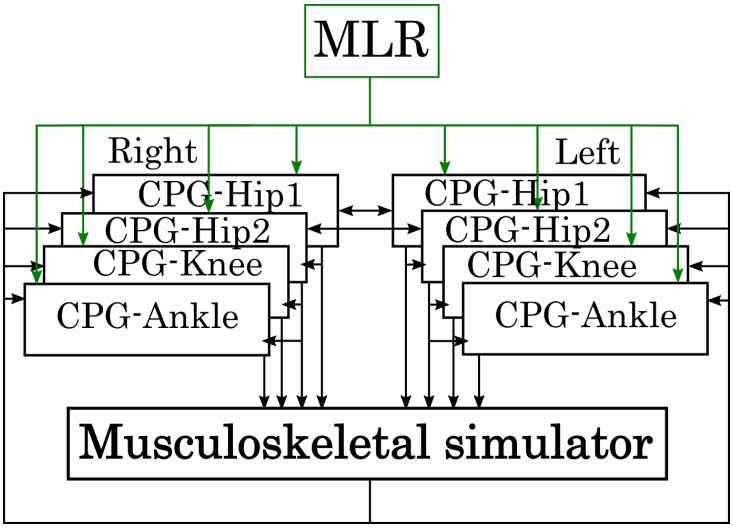


Figure 5.10: Arrangement and connections of 8 CPGs in the controller.

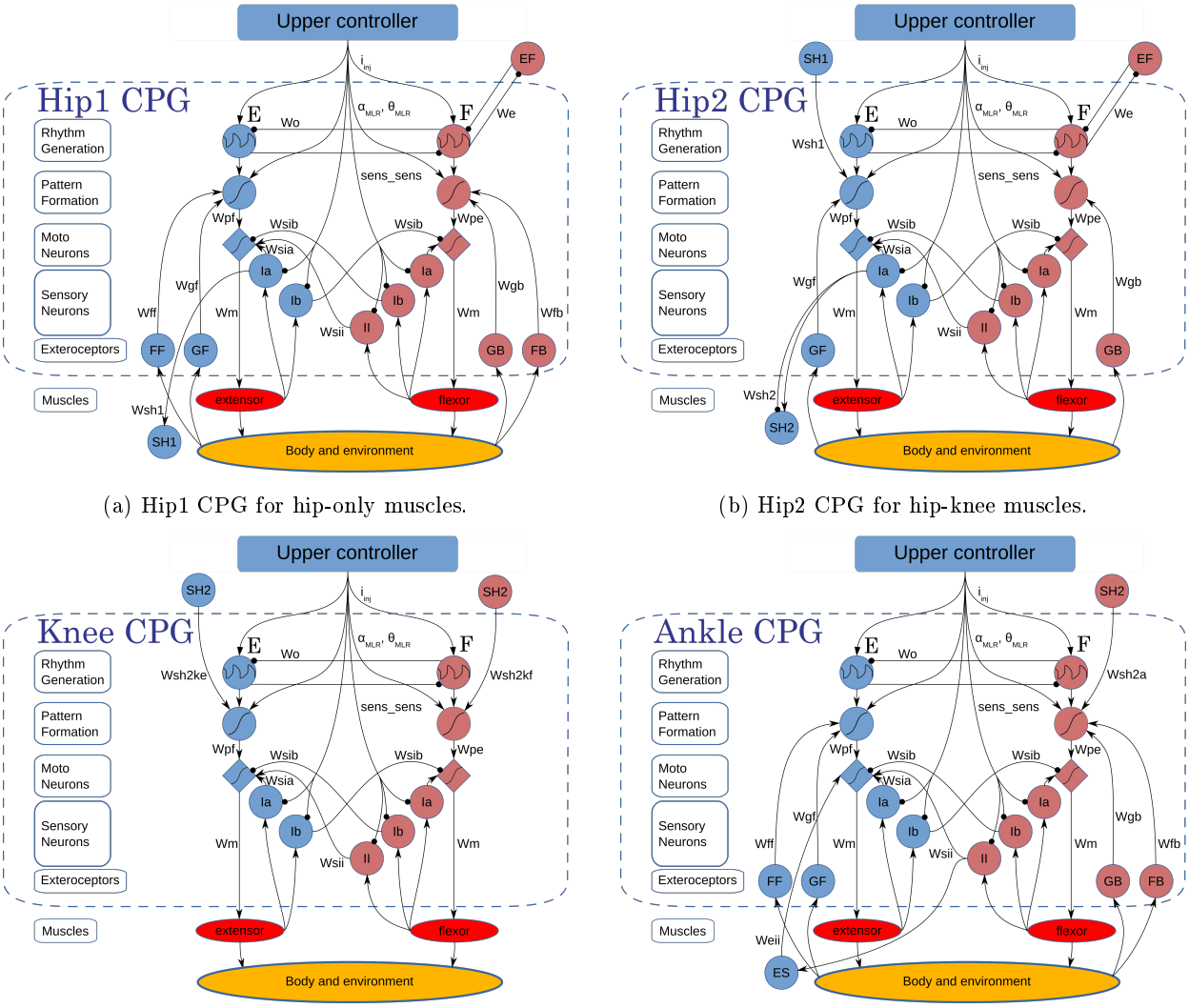
CPG types are mostly similar, with some difference in sensory and interconnections. Hip1 CPG (Fig. 5.11a) has it's RG of flexion half-center interconnected with the other leg's CPG. The connection is inhibitory and is denoted as virtual EF neuron with We weight. Virtual means that there is no additional neuron that transforms the transmitted signal, it is passed directly between RG neurons. Wo weight denotes connection between RG half-centers inside CPG. RG neurons have the same initial frequency, but different initial phase to account for normal muscle activation timing from FA phase (see Section 4.1).

Then, PF layer has an input from trunk angle sensor FF and FB neurons as well as ground pressure GF and GB neurons. $\cdot F$ in them denotes trunk leaning forward or pressure on the front part of foot, i.e. toes. Similarly, $\cdot B$ is trunk leaning backwards or pressure on the back of foot, i.e. heel.

MN layer integrates all proprioceptive information. The connections shown here are under discussion; the a priori idea to apply all proprioceptive sensors in CPG controller has an alternative. Further described phase-oriented controller architecture divides proprioceptive sensors between rhythmic CPG controller and feedback-based balance controller as in Section 2.5.

Nevertheless, sensory neurons (SN) used here are of three types: Ia, Ib, and II, having different roles in the model, following Section 2.0.2. SN Ia (with weight $Wsia$) reacts to muscle's velocity of contraction, which is maximal during opposing motoneuron impulse, and doesn't allow too big angular velocity. SN Ib (with weight $Wsib$) react on muscle's built up force. It inhibits opposing motoneurons, ensuring the order of muscle stretches. Finally, SN II is present only on flexion side and excites only extension motoneuron (Fig. 2.2). SN II (with weight $Wsii$) activates while flexor muscle's length is maximal thus preserving the stability of gait.

As for the target, Hip1 CPG controls flexion and extension of hip-exclusive muscles in sagittal plane. This CPG uses both exteroceptive signals: ground and vestibular. And they both are connected to PF layer with their respective weights (Fig. 5.11a). Ia-type receptor of Hip1 CPG from flexion side is also connected to flexion PF neuron of Hip2 CPG to propagate synchronicity from top CPG to bottom ones as described from biological evidence [54].



(a) Hip1 CPG for hip-only muscles.

(b) Hip2 CPG for hip-knee muscles.

(c) Knee CPG for knee-ankle muscles.

(d) Ankle CPG for ankle muscles.

Figure 5.11: Each CPG detailed for one leg. Lines ending with arrow and circle stand for excitatory and inhibitory connections respectively.

Hip2 CPG (Fig. 5.11b) controls muscles that affect both hip and knee joints. It is similar to Hip1, except of vestibular neurons FB and FF and added $SH1$ virtual neuron with $Wsh1$ weight from Ia-type flexion sensor of Hip1 CPG. Again, virtual means that there is no additional SN neuron that transforms SN Ia output, this signal is passed directly to PF layer.

Hip2 CPG uses ground force sensor and muscle sensor from previous CPG ($SH1$). They both are connected to PF layer with their respective weights. Ia-type receptor of Hip2 CPG from flexion side is propagated to both Knee and Ankle CPGs in a more complex manner.

Knee CPG (Fig. 5.11c), has the least amount of sensors and connections. It doesn't have ground or vestibular connections, but it has two connections from Hip2 CPG, denoted as $SH2$ virtual neuron. It adds an excitation connection to flexion-side PF neuron with weight $Wsh2kf$ and an inhibition connection to extension-side with $Wsh2ke$, as described in biology [54].

Knee CPG controls muscles of knee joint as flexor and ankle joint as extensor. Mainly, Gastrocnemius muscle affects both knee and ankle joints. The other knee muscles are monoarticular (affect only knee joint). This asymmetry increases the complexity of control and is overcome by co-contraction and synergy of muscles.

Ankle CPG terms dorsi-flexion of ankle joins as flexion and plantar-flexion as extension (Fig. 5.11d). It also has connections to ground and vestibular sensors, and to excitation virtual neuron $SH2$ from Ia-type sensor of Hip2 CPG with weight $Wsh2a$ only to flexion PF of Ankle CPG.

In the following, simulation results with simpler GAIT2DE simulator will be demonstrated. After that, an implementation for an extended set of muscles with OpenSim simulator will be described and illustrated.

5.3.1 Simulation with GAIT2DE model

This section shows simulation of gait control using the controller with 8 CPGs, one for each pair of antagonistic muscles. Thus, it uses all sixteen muscles of GAIT2DE model. There are eight muscles per leg divided into four pairs.

General scheme of the controller and its connections is shown on Fig. 5.12. Connections from the model to controller (in green) denote two entities. Ground pressure sensing affecting Hip1, Hip2, and Ankle CPGs. And in-series connection between CPGs for joint synchronization.

Hip1 CPG controls Iliopsoas (ILI) and Gluteus (GLU) muscles, which affect only hip joint, former being denoted as flexion and latter as extension. Hip2 CPG controls Rectus Femoris (RF) and Hamstrings (HAM) muscle that affect both hip and knee joints and denoted by their hip function. Knee CPG controls Gastroc (GAS) and Vastus (VAS) muscles. Although the latter extension muscle affect only knee joint, the Gastroc muscle affect both knee and ankle joints and is put in this CPG as symmetric to Vastus. Finally, Ankle CPG is simpler: Tibialis Anterior (TA) for joint flexion and Soleus (SOL) for extension.

The parameter values were searched for with Genetic optimization algorithm (see Chapter 4.3.1) and are given in the Annex Table A.1. The simulation has produced the plots of Fig. 5.13. They show signals in all 8 CPGs during 10 seconds of gait simulation. We can see that RG phases are the same in Hip1-Hip2 and Knee-Ankle CPGs and, while fixed for the sake of parameter search simplicity, they show the possibility to reduce the number of CPGs in controller. But the resulting muscle excitation from MN layer is different in all CPGs, accounting for different muscle and joint configurations. These facts

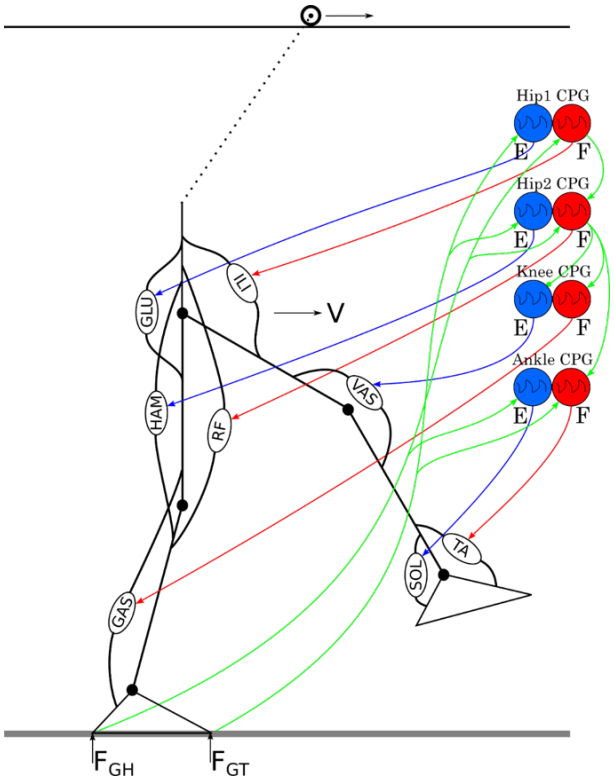


Figure 5.12: Structure of GAIT2DE model with CPG connections. Muscles are shown for one leg.

must be acknowledged while simplifying CPG controller architecture. Afferent feedback noticeably affect control only in Ankle CPGs, where the ground feedback is most important.

Fig. 5.14 shows one gait cycle extracted from the simulation animation. From the initial standing position of biped model, a stable cycle starts after four seconds of walk. During simulated walk the most significant were *Ia* and *Ib* proprioceptors, except for Hip2 CPG. *II*-type sensor had almost the same state during whole gait cycle. The animation itself can be found in folder `Thesis animations`[9] as `Gait2de 8CPG gait`.

The controller with the found set of parameters result in slow stable gait with wide step. Different parameters allow to change resulted type, speed of gait and stride length. Parameters of each CPG can be varied independently to simulate different types of gait, from slow, narrow steps observed in parkinsonian patients (`Gait2de slow gait`); to wide jump-like jogging gait (`Gait2de fast gait`).

5.3.2 Simulation with OpenSim model

In this section, the gait simulation results are described using the same controller architecture, but a different musculoskeletal model. Two types of gait were simulated: normal gait and gait affected by sudden stop and release of upper control signal, emulating basal ganglia decision-making behavior.

Fig. 5.15 shows CPG connections with OpenSim model. Each CPG controls two muscle groups, blue arrows point at flexion muscle groups and CPG half-centers and red ones respectively point at extension

[9] `https://mega.nz/#F!sx1hyBIb!WmACHs6ZVGNt08MzmzpZfA`

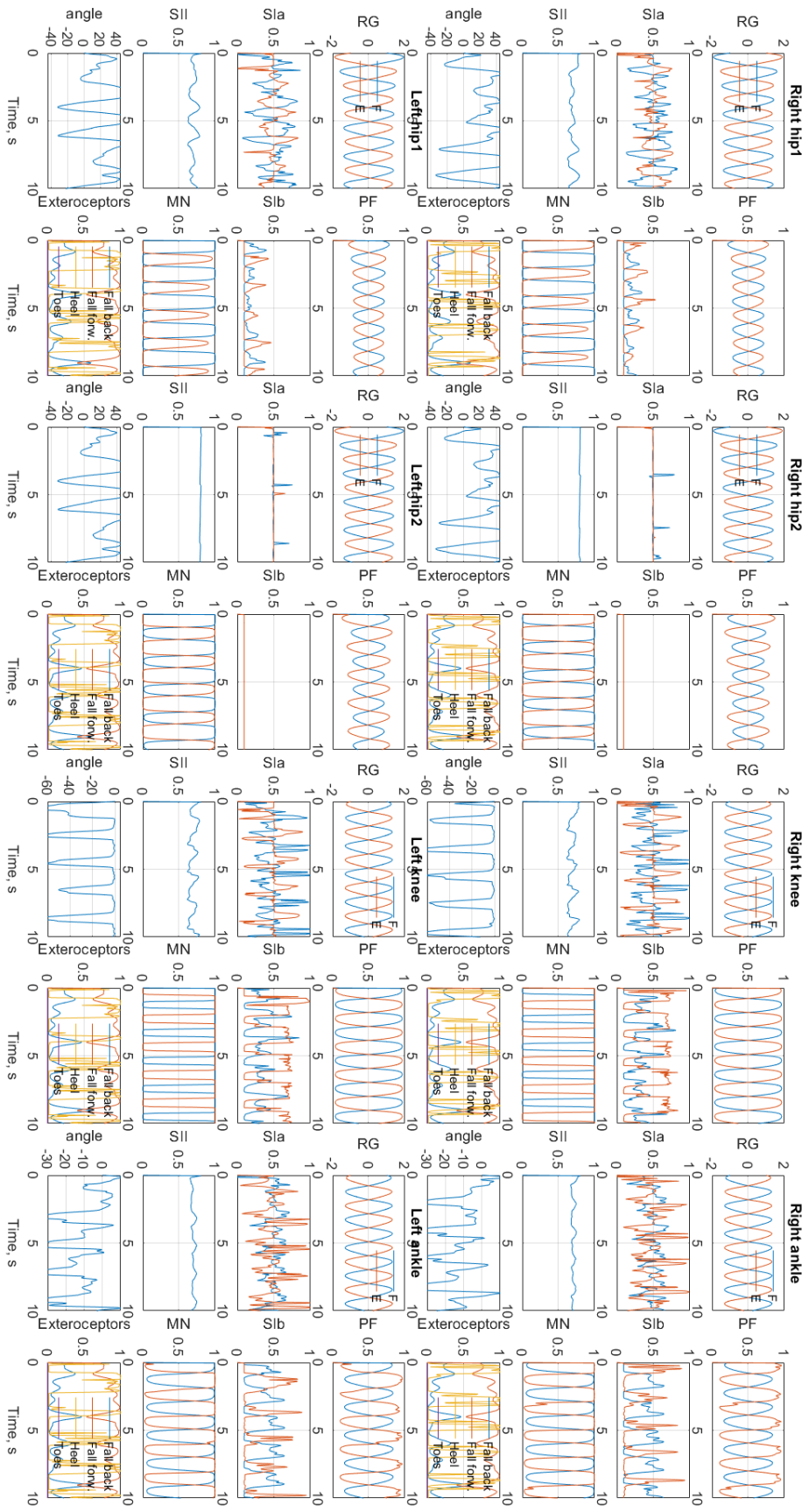


Figure 5.13: Signals of all 8 CPGs during gait simulation.

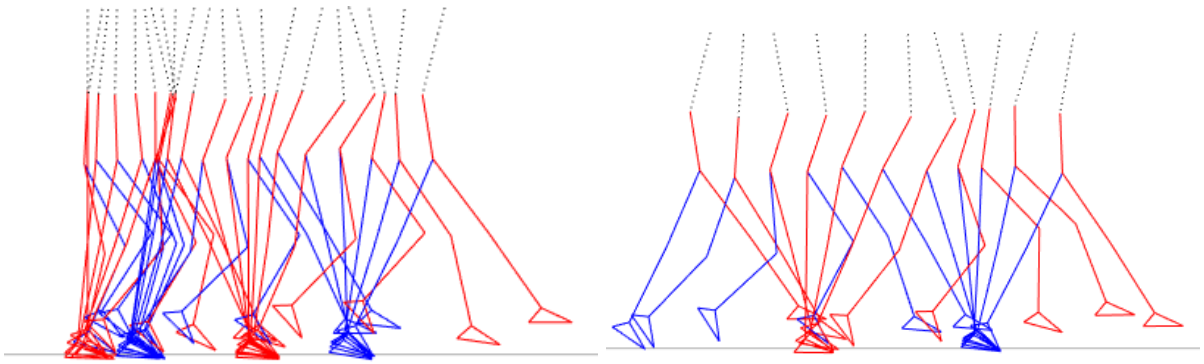


Figure 5.14: Consecutive frames of stable gait in Gait2de simulation, time between frames is 0.2 s. Left: initial transient gait from standing phase to walking. Right: further stable gait.

groups that exchange efferent excitation and afferent feedback. Grey arrows denote sources and receptors of exteroceptive feedback of body angle and ground reaction force. Green arrows denote unidirectional inter-CPG connections.

In OpenSim model, CPG network controls more muscles that are divided in the same eight groups. Hip1 CPG controls flexor group (H1F) consisting of Iliacus and Psoas muscles and extensor group (H1E) consisting of three Glutei muscles: Maximus, Medius, and sagittal-affecting portion of Minimus muscle.

Hip2 CPG controls flexor group (H2F), consisting only of Rectus Femoris muscle, which also acts as extensor for the knee. Similarly, extensor group (H2E) consists of Biceps Femoris Long Head, Semimembranosus, and Semitendinosus muscles, which extend for hip and flex knee.

Knee CPG's flexion muscle group (KF) has only Gastrocnemius, which affect both knee and ankle joints. The extensor group of Knee CPG (KE) consists only of knee-affecting muscles: Vastus Intermedius, Lateralis, and Medialis.

Ankle joint dorsi-flexion muscles compose flexion group of Ankle CPG (AF). They are Extensor Digitorum Longus, Extensor Hallucis Longus, and Tibialis Anterior muscles. Plantar-flexion muscle of extension muscle group (AE) is just Soleus.

Simulations with OpenSim perform much slower than with GAIT2DE due to increased complexity of the model and simulation platform. Thus, Genetic optimization was considered inapplicable because of numerous generations' calculation. Instead, BlackBox method was used (see Chapter 4.3.1).

To illustrate the capabilities of gait simulation platform, two types of walking gait: normal and temporarily interrupted by MLR signal are shown. They both use the same musculoskeletal model and same set of CPG network settings and parameters (see Annex Table B.1).

5.3.2.1 Normal gait

The normal human walking gait was simulated during 6 seconds to show the transition from standing up in double support phase to walk. The animation is called `OpenSim 8CPG normal gait`[10]. Fig. 5.16 demonstrates external signals of the model: SNF as trunk angle sensor and SNG as ground reaction sensor. On SNG plot, standard M-shape of ground force reaction is appeared due to stepping. Individual

[10] `https://mega.nz/#F!sx1hyBIb!WmACHs6ZVGNt08MzmzpZfA`

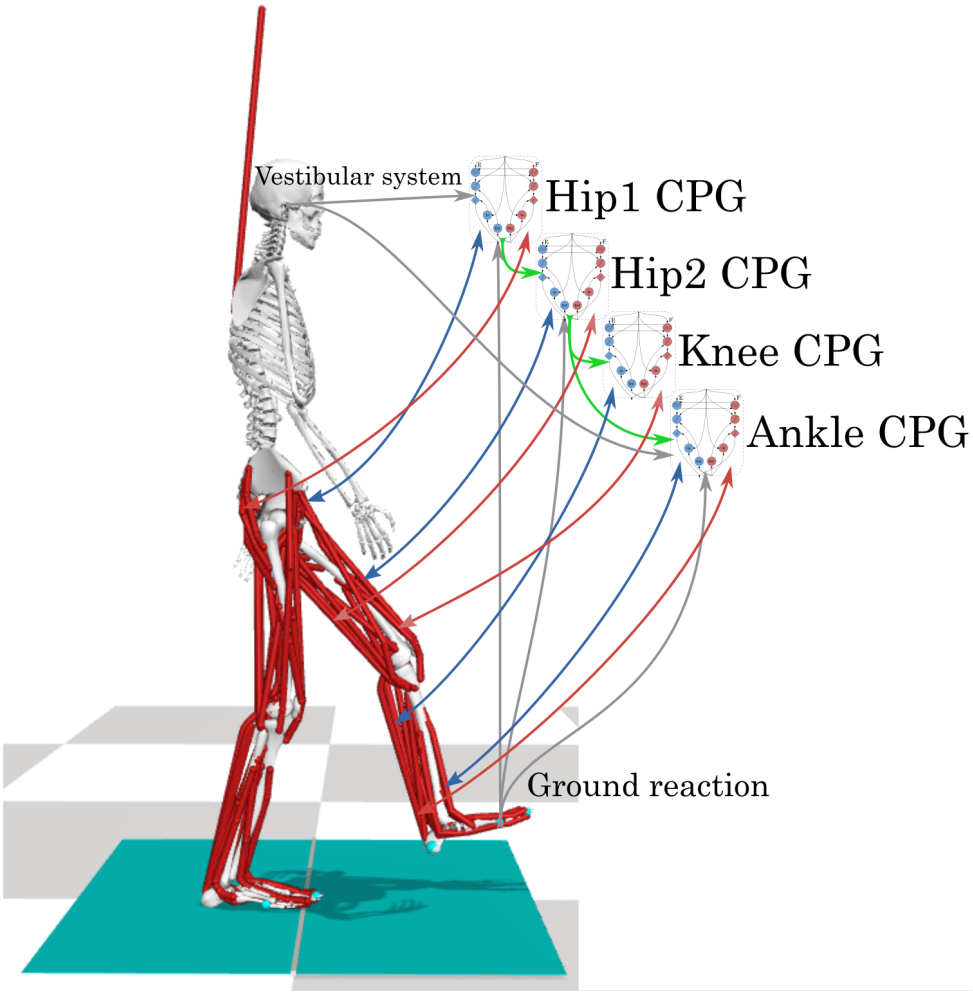


Figure 5.15: Structure of CPG connections with OpenSim model.

steps can be distinguished on sections 0.5 to 1 s; 1.8 to 2.5 s; 3.5 to 4.2 s; 5 to 6 s.

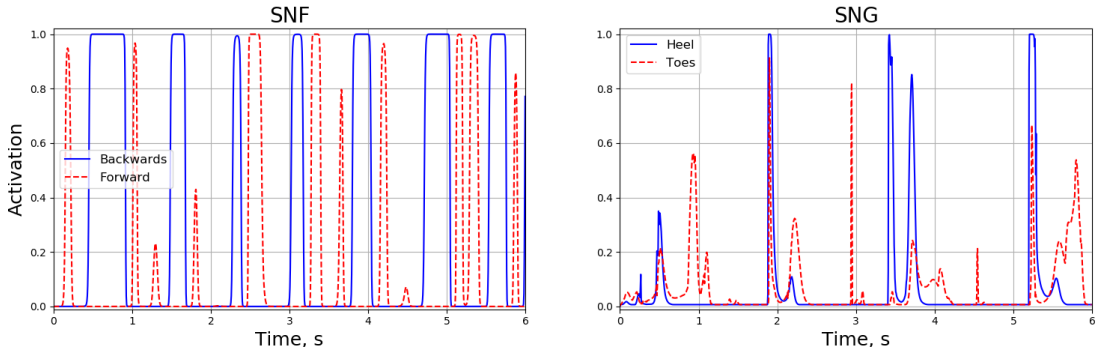


Figure 5.16: SNF is trunk angle sensor and SNG is ground force sensor of right leg.

Fig. 5.17 shows signals in CPGs of right leg (signals in left CPGs are similar and therefore not given). Lines marked as F (in solid blue) and E (in dashed red) stand for flexion and extension respectively.

Muscles' resulting activation from motoneurons of CPGs is shown on Fig. 5.18 along with the names of muscles of each group. These groups receive individual excitation and result in overlapping activation plots for muscles in group.

Muscles' actuation enables the model to walk. The joint angles are shown on Fig. 5.19, along with default human values denoted in red dashed line. Very brief negative knee angle values, appeared on plots, do not affect the gait and are the result of energy dissipation in OpenSim for the case of joint angle limits, and are not observed on simulation animation.

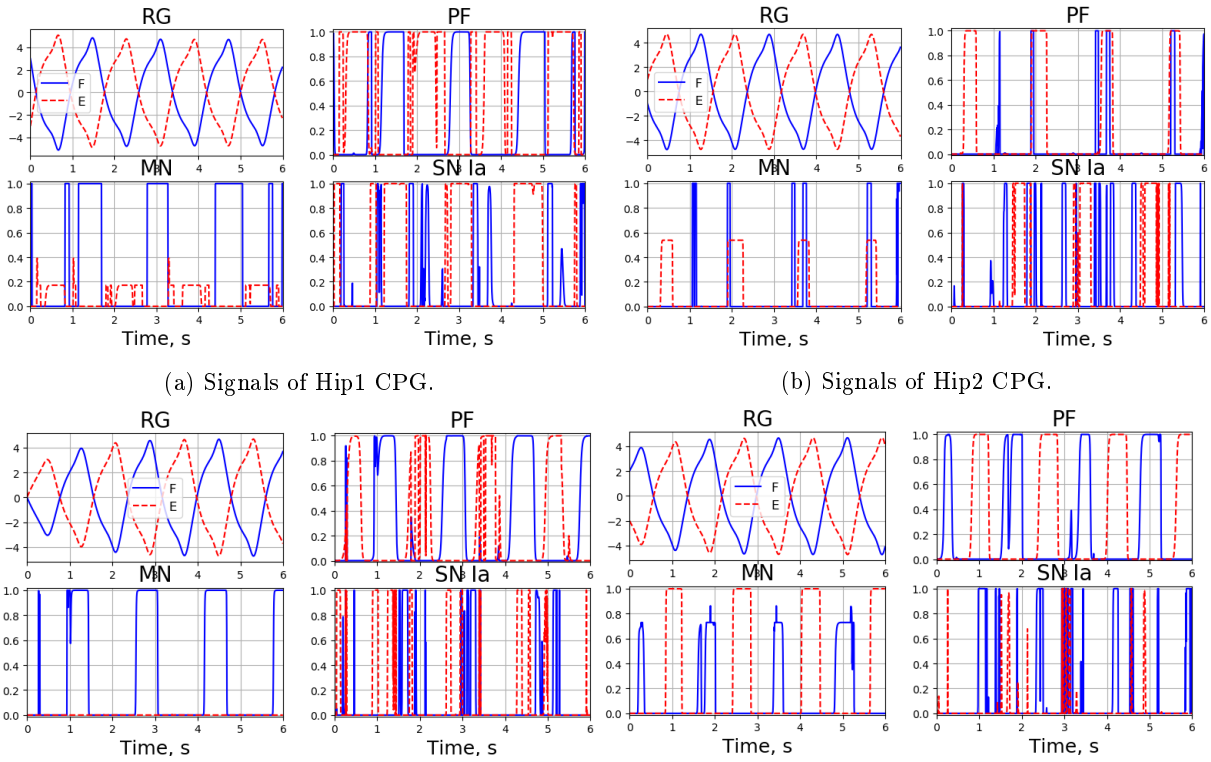
The shape of simulated joint angles in hip and knee is largely similar to human angles, although ankle joint is different. Higher amplitude of hip angle during gait was required because of ankle structure and control, to remove model's stumbling to ground. Difference between human ankle behavior and simulated one comes from the difficulty to produce a realistic model of the foot (number of muscles, sole flexibility) and thus to simulate correctly the foot/ground contact. In the OpenSim model, the foot's sole is rigid and creates some changes in the joint angle behavior compared to the human [109]. This difference is observed in biped robotics as well, because robot feet are rigid comparing to human's.

Fig. 5.20 shows two characteristics of the gait: pelvis height and horizontal velocity. Pelvis height ranges for 10 cm due to stepping, while human values' range is about 8 cm [111]. Horizontal velocity of pelvis and its moving average show the transition from standing position to walk during first 2 seconds.

Fig. 5.21 shows several frames of simulated walking gait of second gait cycle in OpenSim. Time between frames is 0.2 seconds. Color of muscles denote their activation, from blue as inactive to red as fully active. On top, connected to model's back, is supportive harness-elastic that keeps the model upwards in the absence of balance control.

5.3.3 Disrupted gait

During this simulation, the upper neuronal control level is enabled. It acts like a simple decision-making module that controls start/stop of gait and can switch its mode between walk/run. This function is presented as a threshold signal that inhibits and releases motor function at defined time. The animation



(a) Signals of Hip1 CPG.

(b) Signals of Hip2 CPG.

(c) Signals of Knee CPG.

(d) Signals of Ankle CPG.

Figure 5.17: Signals of three CPG layers (RG, PF, MN) in right leg. SN *Ia* is muscle stretch velocity sensor.

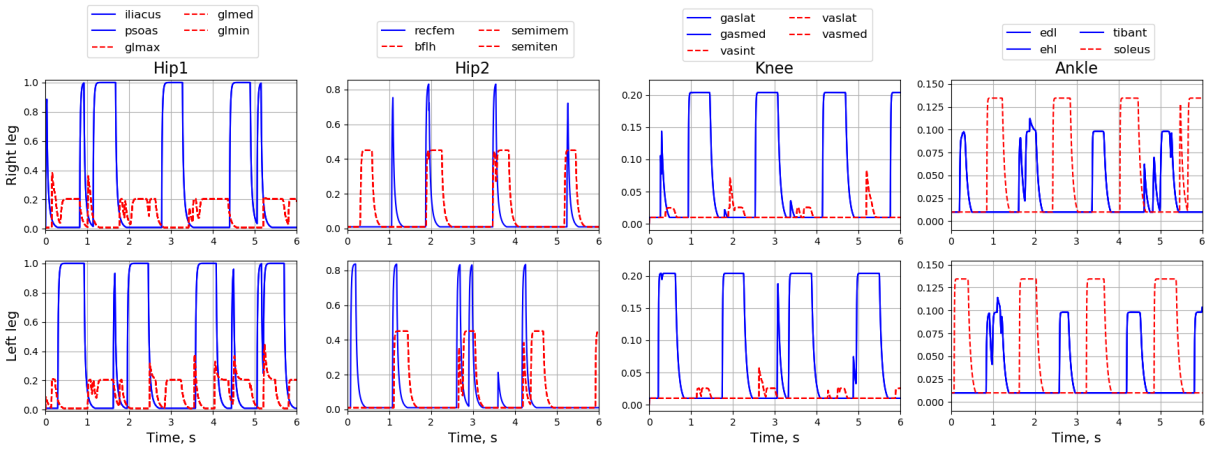


Figure 5.18: Muscle activation denoted by group composition on top.

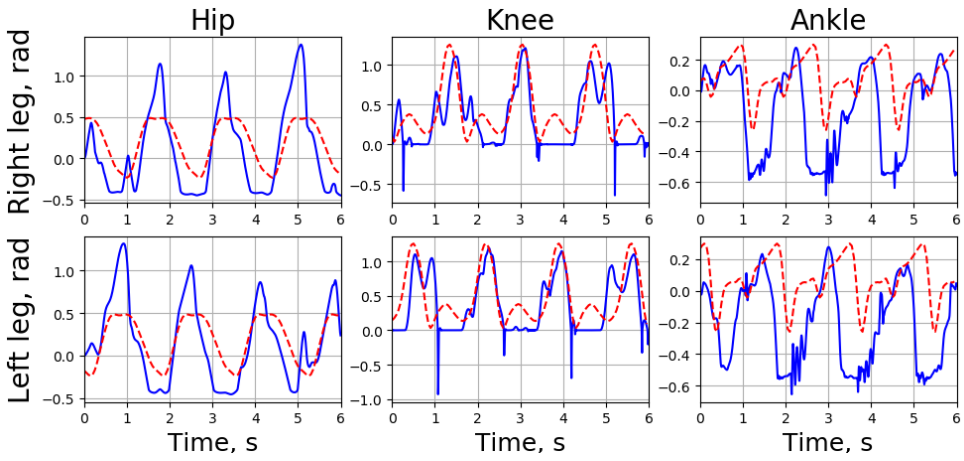


Figure 5.19: Joint angles during normal gait. Dashed lines are the normal human joint angles, extracted from [4].

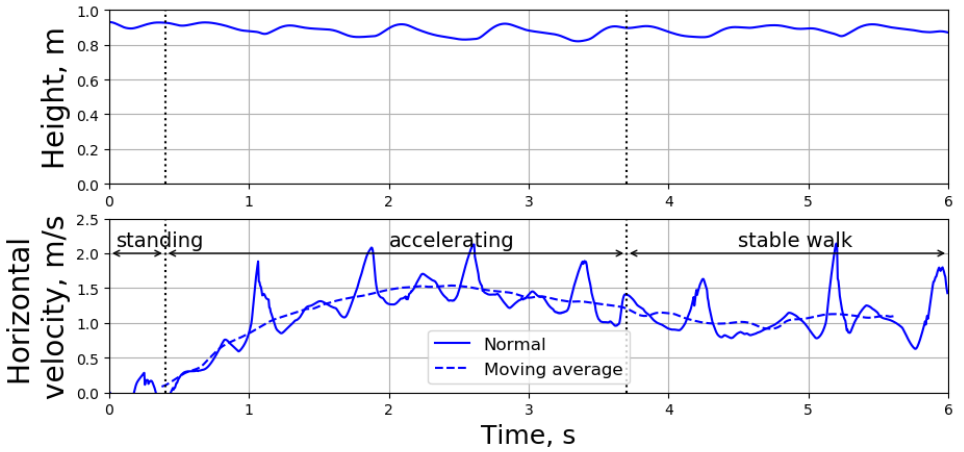


Figure 5.20: Top: pelvis height. Bottom: pelvis velocity. Dashed line shows moving average of velocity with window of 0.75 seconds.

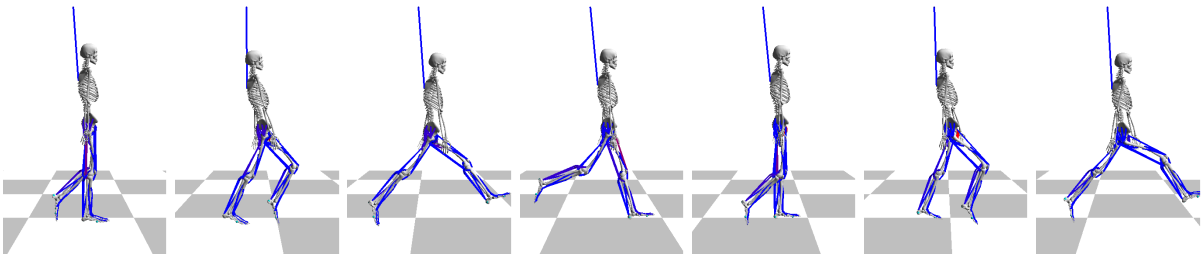


Figure 5.21: Frames of gait simulation, every 0.2 s.

is called `OpenSim 8CPG gait disrupted at 4s`[11].

Abnormal gait is simulated by manipulating the θ_{MLR} signal that clamps CPG control from PF layer and further. This gait was simulated for 12 seconds to show model's motor function stop and release. $\theta_{MLR} = 5$ between 3^{rd} to 7^{th} second. The rest of time $\theta_{MLR} = 0$.

The absence of signal from PF layer and further in motoneurons results in no muscle activation (Fig. 5.22). In this state, only the supportive harness-elastic prevents model from falling due to lack of reflexes and vestibular system. Top end connection of harness moves above model's center of mass. During the disrupted state, joint angles follow the overall body position provided by support. As relaxed body tries to fall backwards, hip angle stays on 0.5 radiant, knee extends to straight posture, and ankle is plantar-flexed (Fig. 5.23).

After the release of CPG signal on 7^{th} second, controller output resume to feed the oscillation and restarts gait, that is shown on Fig. 5.24. It's important to note, that normal gait stumbles on 6^{th} second with this parameter set. Disrupted model moves by inertia for a second before stop and for another two seconds, model gains velocity until the average of 1 m/s, defined by CPG frequency and weights.

This effect of θ_{MLR} resembles a freezing of gait phenomena, observed in parkinsonian patients. Indeed, Parkinson's is a neurodegenerative disease that originates in basal ganglia region that innervates MLR. Thus, the simulation presented here can be seen as the first step towards a more complex modeling platform of cerebral-spinal links.

Nevertheless, it can be observed that during freezing of gait the modeled human body cannot support itself for standing up and the elastic acts like a real medical harness. This problem could be avoided by adding to each CPG specific neurons able to produce tonic activity for muscles in order to control the standing posture and balance. Such reflexive controller is described in Chapter 2.5.

5.4 Phase-targeted CPG controller

The muscle-targeted CPG architecture, in the case of increasing model complexity, expands into a network with a number of CPGs equal to the number of muscles. But if a controlling accuracy stays behind the model's simulation accuracy, a number of CPG stays the same. These CPGs, regardless of their muscle's function, can thus be approximated by one CPG. This CPG therefore will have a unique activation phase and control a group of muscles by their timings rather than function.

And at the same time, at-glance investigation of muscle activation EMG plots (see Section 4.1) reveals rough separation of muscle by their activation timings. The sagittal muscles used with OpenSim model assume four muscle activation phases with two non-spanned pairs. This assumption lays in the phase-targeted CPG controller architecture. The idea is to use just two CPG per leg, each able to control a non-spanned activation phase.

The first gait phase lasts from mid-swing to terminal stance, when hip and knee flexion muscles are active. During the rest of the gait cycle, antagonistic extension muscles are active. The second phase, active from per-swing to load response, is suggested to hots activation of knee and ankle flexion muscles. As before, the rest of the time, knee-ankle extension muscles should be activated.

This section describes such controller with 4 CPGs (Fig. 5.25). The CPGs are similar to ones in

[11] `https://mega.nz/#F!sx1hyBIb!WmACHs6ZVGNt08MzmzpZfA`

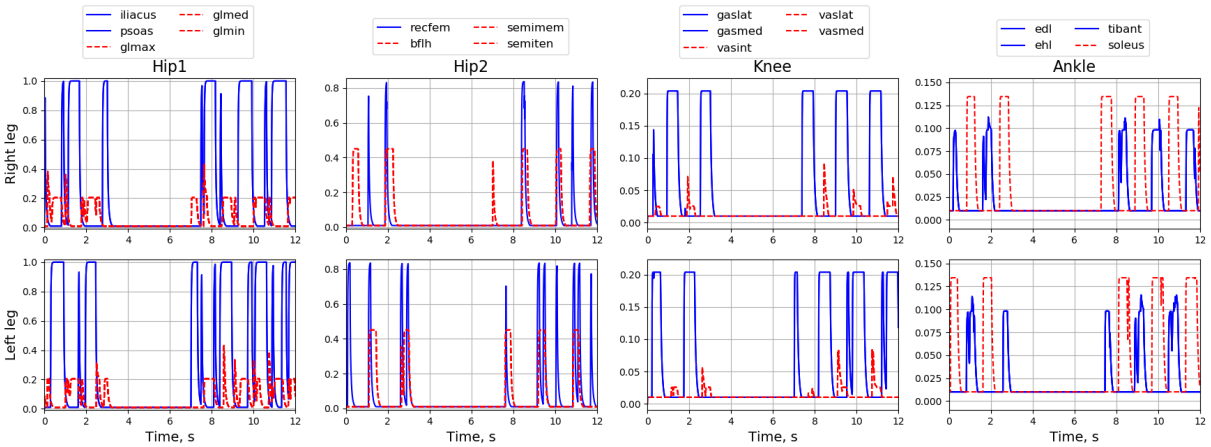


Figure 5.22: Muscle activation in disrupted gait. θ_{MLR} signal clamps CPG control from 3^{rd} to 7^{th} second.

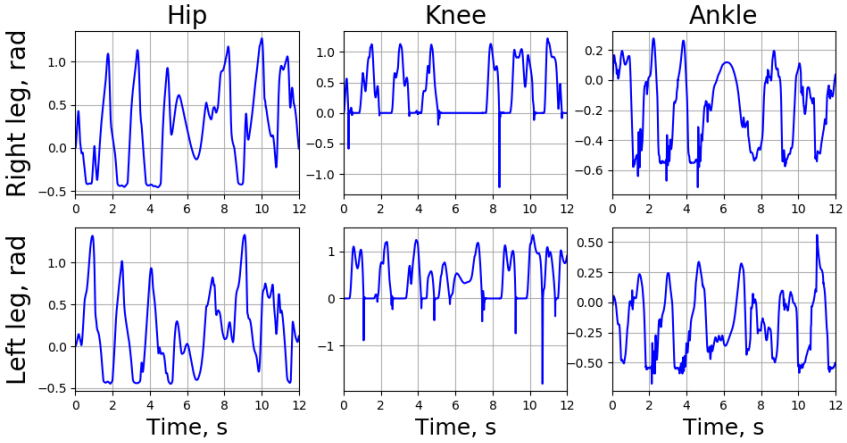


Figure 5.23: Joint angles during disrupted gait. θ_{MLR} signal clamps CPG control from 3^{rd} to 7^{th} second.

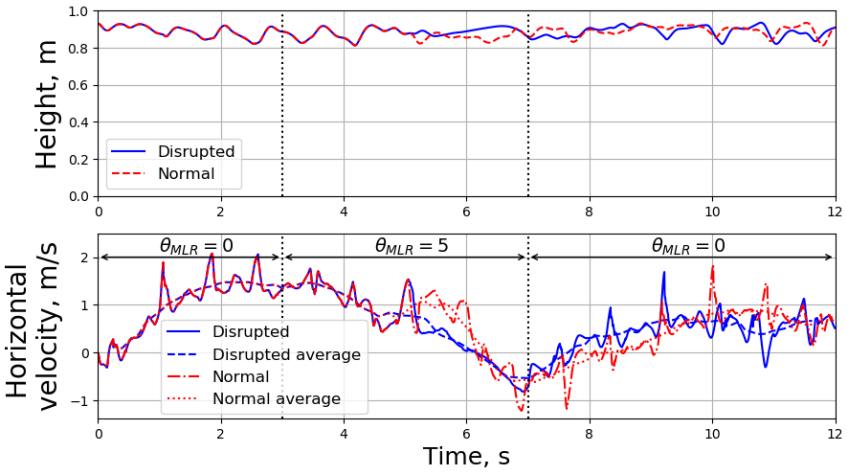


Figure 5.24: Normal and disrupted pelvis height and velocity with moving averages.

Section 5.3 with OpenSim simulator, but the network of them is simpler as there is only one in-series connection, from Hip-Knee CPG (HK) to Knee-Ankle CPG (KA). The used names of CPGs, Hip-Knee and Knee-Ankle, briefly describe their target muscles, but don't limit the muscles in each group.

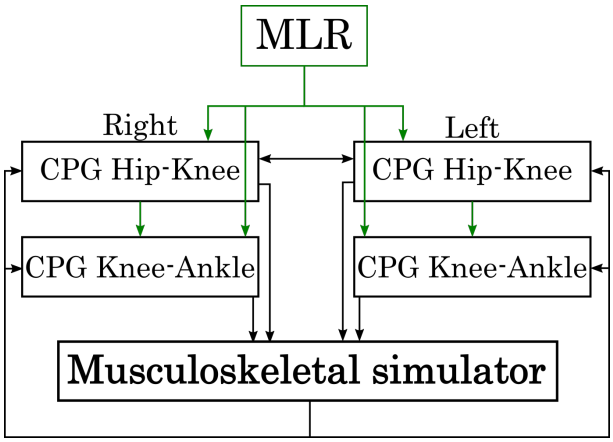


Figure 5.25: Arrangement of 4 CPGs in controller.

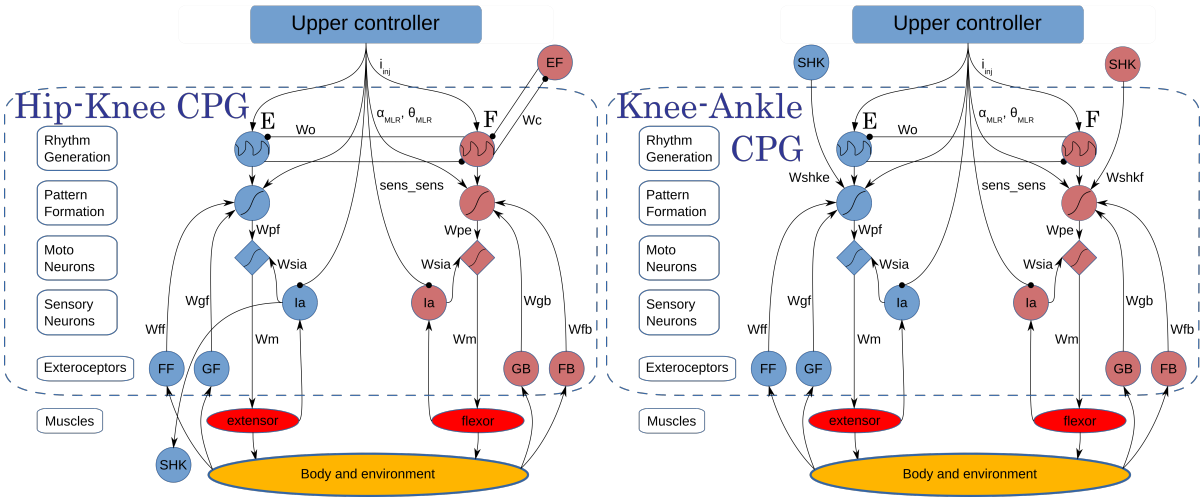
This approach is simpler than muscle-targeted controller and it allows easier parameter search due to halved number of CPGs, but it is less flexible by the means of control phases. Additional control flexibility may be achieved by muscle synergy approach, touched in the Discussion. Briefly, synergy means that one muscle may receive control signals from more than one CPG, which is not the case for described controllers.

In current architecture, the reflexive balance controller (see Section 2.5) is added to the simulation to help the model maintain equilibrium. It receives input from upper controller as $sens\_sens$ signal, which inhibits balance reflexes during gait. It utilizes afferent feedback of type-Ib and II, while CPG controller uses only type-Ia muscular feedback. Simulation that uses only balance controller with details on its functioning, is described in Section 5.5.

As in previous models, each CPG receives input from upper controller denoted as MLR. Fig. 5.26 shows connections inside both CPGs. Input constitutes of i_{inj}, α_{MLR}, and θ_{MLR} signals. In current simulation, gait mode switch with i_{inj} is not used, as well as α_{MLR} for gait speed. θ_{MLR} and $sens\_sens$ were used similarly, to transit from standing to walk and back to stop during simulation. At start, balance controller is active and CPGs are inhibited. Then balance controller is inhibited through $sens\_sens$ and θ_{MLR} releases CPG controller.

Initial phase of RG oscillation was determined from experimental muscle activation phases (see Chapter 4.2) and desired part of gait cycle (see Chapter 4.1): FA phase. HK CPG, as before, synchronizes with the other leg's HK through virtual neuron EF with Wc weight (Fig. 5.26a). Both CPGs now connect to ground and fall exteroceptors. The weight values of this controller are given in the Annex Table B.2. As before, BlackBox was used to search for optimal controller parameters. Similarly to muscle-targeted CPG controller, CPG HK gives its muscle type-Ia afferent to CPG KA as SHK virtual neuron with weights $Wshkf$ and $Wshke$.

Each CPG's motoneurons control individual group of muscles, noticed to be active simultaneously. CPG HK flexion half-center excites Biceps Femoris (Short Head), Gracilis, Iliacus, Psoas, Rectus Femoris,



(a) Hip-Knee CPG for hip and knee muscles.

(b) Knee-Ankle CPG for knee and ankle muscles.

Figure 5.26: Both CPG detailed for one leg. Lines ending with arrow and circle stand for excitatory and inhibitory connections respectively.

and Sartorius muscles. Its extension half-center controls Adductor Magnus (Ischiocondylar Portion), Biceps Femoris (Long Head), Gluteus Maximus, Semimembranosus, Semitendinosus, Tensor Fascias Latae, and Vastus muscles (Intermedius, Lateralis, and Medialis). CPG KA during flexion phase excites Extensor Digitorum Longus, Extensor Hallucis Longus, and Tibialis Anterior muscles. And its extension part controls Flexor Digitorum Longus, Flexor Hallucis Longus, Gastrocnemius (Lateral and Medial Heads), Peroneus Brevis, Peroneus Longus, Soleus, and Tibialis Posterior muscles.

The excitation is passed to OpenSim simulator, which returns feedback both muscular and environmental. Muscle feedback of *Ia*-type is used in CPG controller to limit joint angular velocity. Feedback of *Ib* and *II*-type are passed to balance controller.

We have performed a simulation of initiation of gait from standing phase through inhibiting CPG controller and releasing balance controller and further walking. The animation can be found as **OpenSim 4CPG gait**[12].

Fig. 5.27 shows signals from the part of CPG controller with RG, PF layers and exteroceptors. First column of plots shows outputs of RG layer, each having a different starting phase, although starting phases of RG in right and left legs are opposite. θ_{MLR} value from the upper controller clamps CPG controller at the start of simulation. θ_{MLR} affect PF layer, the last column of plots. In-between, there are body angle and ground reaction force values with corresponding sensory neurons.

Fig. 5.28 shows signals from the part of CPG controller with SN *Ia* and MN layer. SN *Ia* is intended to affect gait in extreme conditions, when angular velocity of a joint is too high. Thus, their effect is not noticeable in this simulation. MN plots column show the absence of controller output at the start of simulation, while muscle excitation combine output from both CPG and balance controllers.

Fig. 5.29 show joint angles during simulation as well as muscle activation as a result of CPG and balance controllers. The last joint, MTP (foot toes joint), is not directly controlled. MTP is merged

[12] https://mega.nz/#F!sx1hyBIb!WmACHs6ZVGNt08MzmzpZfA

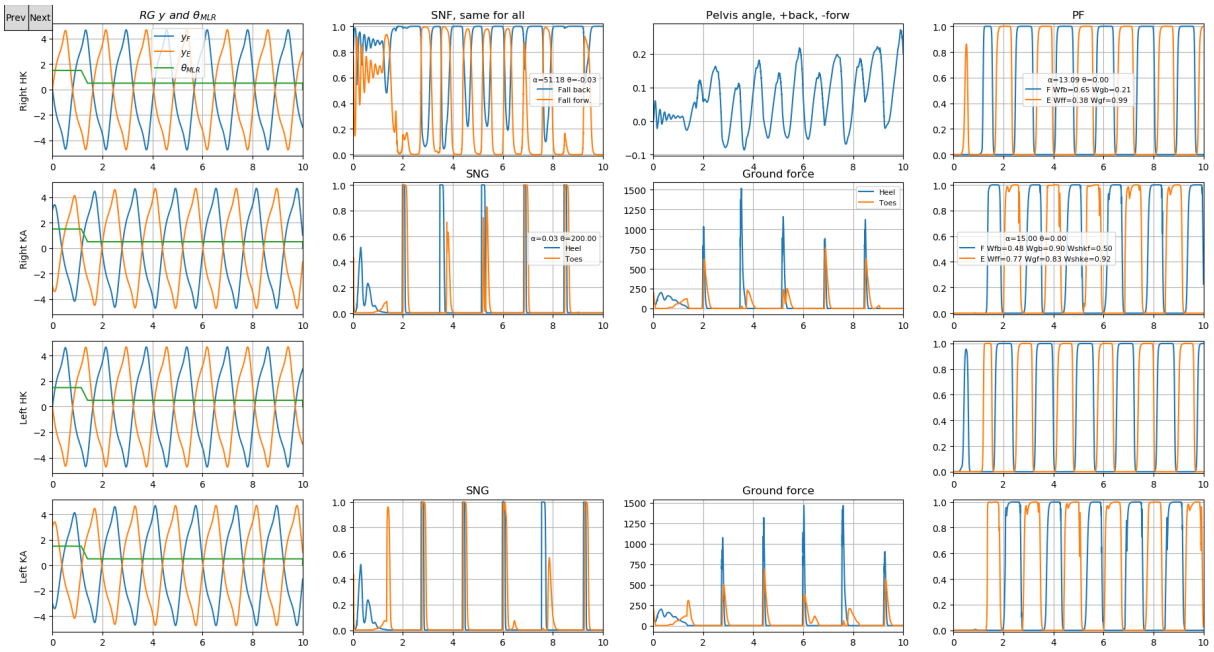


Figure 5.27: Signals from the RG and PF layers and exteroceptors during gait simulation.

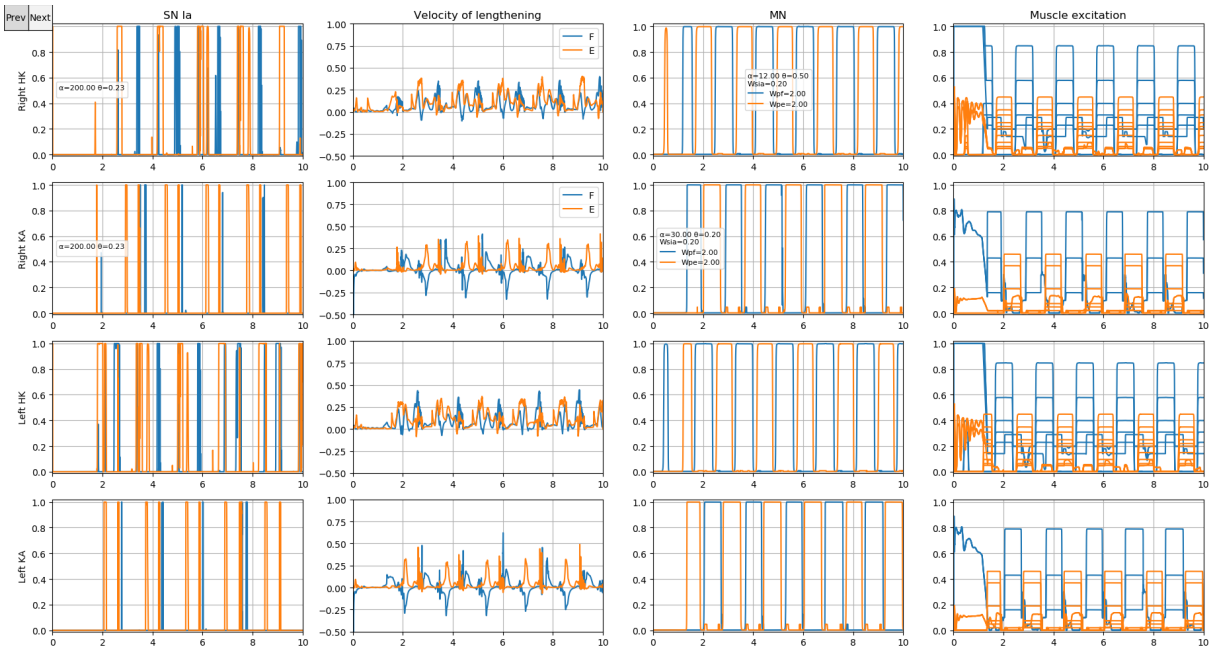


Figure 5.28: Signals from the SN *Ia* and MN layer during gait simulation.

in muscular control with ankle joint and is noticeably affected by energy dissipation issue, described in Section 3.2.

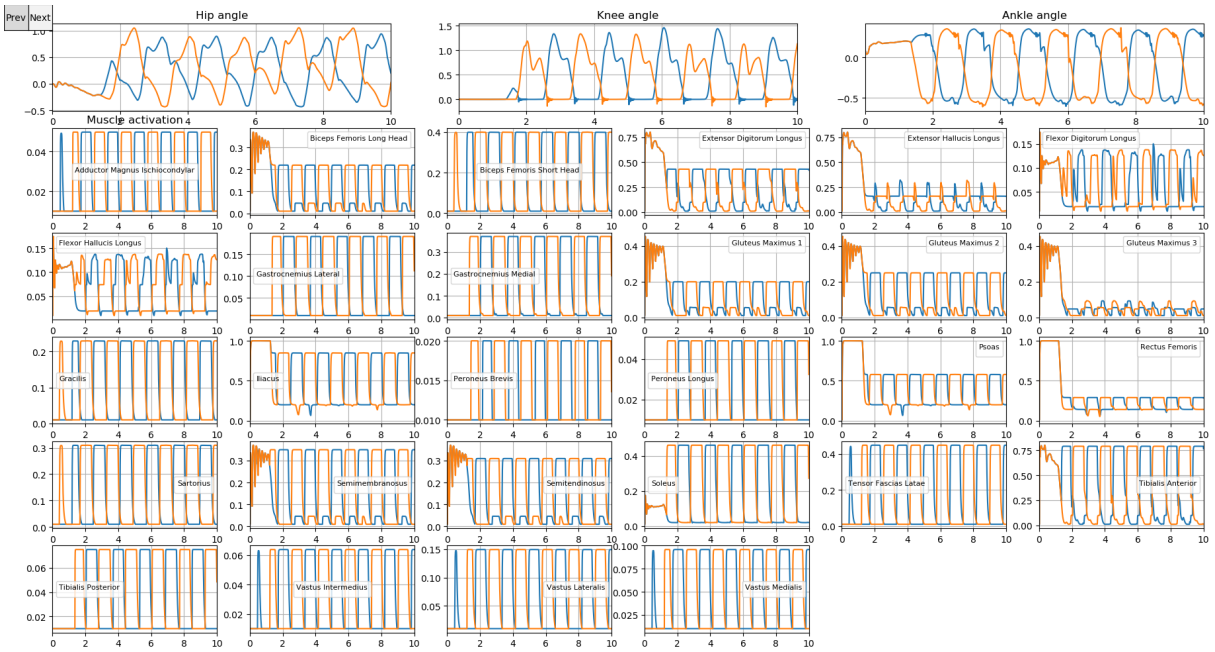


Figure 5.29: Joint angles and muscle activations during gait simulation.

On the muscle activation plots, we can see that different muscle have various role in the simulated gait. These roles are rhythmic gait and balance control. The muscles may be divided into three groups by their role as: mostly rhythmic gait, like Gastroc muscles; muscles with both roles, like Iliacus; and muscles, mostly used by balance controller, like Flexor Digitorum Longus.

Fig. 5.30 shows pelvis height and velocity during gait to demonstrate the stability of gait. We can see, that after releasing CPG control at 1.4 seconds, the model takes time to accelerate till about 3 seconds and then continue to walk at the pace of around 1 m/s. The model slowly decelerates due to falling backwards and hanging on the harness (see **OpenSim 4CPG gait**[13]). The hip height varies around 1 meter as well, cycling periodically.

Fig. 5.31 shows several frames of simulated walking gait in OpenSim. Time between frames is 0.2 seconds. As before, color of muscles denote their activation, from blue as inactive to red as fully active. On top, connected to model's back, is an updated supportive harness-elastic that keeps the model upwards in the absence of balance control. As much work is still need to be done in order to keep the model from falling in the absence of the harness.

5.5 Balance controller

This section demonstrates the maintaining of equilibrium using solely balance controller from Section 2.5, applied to musculoskeletal model in OpenSim simulator.

[13] https://mega.nz/#F!sx1hyBIb!WmACHs6ZVGNt08MzmzpZfA

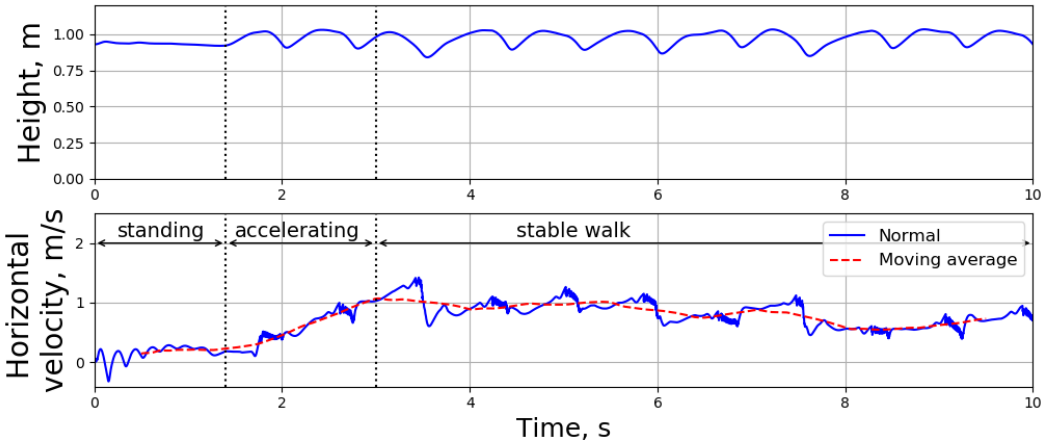


Figure 5.30: Top: pelvis height. Bottom: pelvis velocity. Dashed line shows moving average of velocity with window of 1 second.

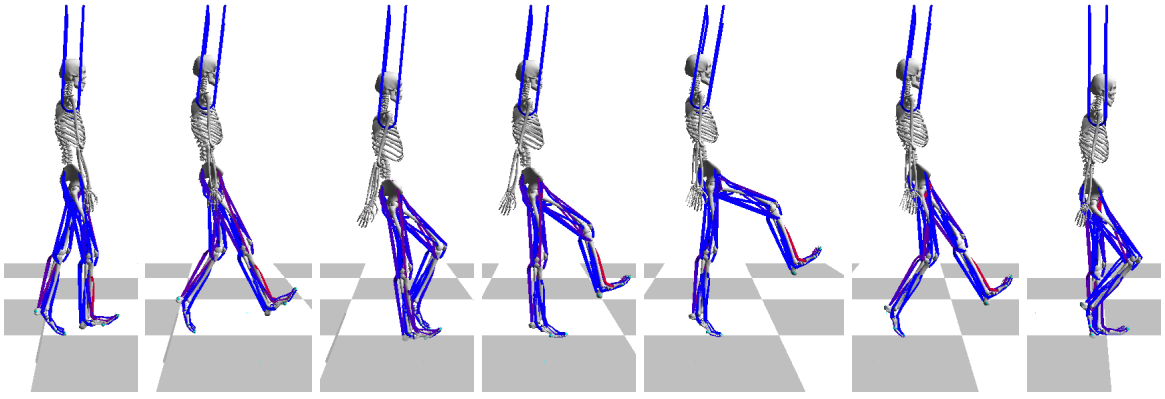


Figure 5.31: Frames of gait simulation, every 0.2 s.

This controller is intended to help maintaining dynamic balance during walk as well as keep balance during stops before and after periods of walking.

The reflex-based balance controller uses seven muscle groups (Fig. 5.32). Muscle groups are: HFL (Iliacus, Psoas), GLU (Gluteus Maximus), HAM (Biceps Femoris Long Head, Semimembranosus, Semitendinosus), RF (Rectus Femoris), VAS (Vastus Intermedius, Lateralis, and Medialis), SOL (Flexor Digitorum Longus, Flexor Hallucis Longus, Soleus), and TA (Extensor Digitorum Longus, Extensor Hallucis Longus, Tibialis Anterior).

Each reflex arc on Fig. 5.32 is independent with its own weight. Muscle groups receive excitation as a sum of inputs from each corresponding arc. Ground (GB, GF) and fall (FB, FF) sensors are shared with CPG controller and use the same parameters. This balance controller is mediated by $sens\_sens$ connection from upper controller that inhibits reflex output during gait. If balance controller is used along with CPGs, the resulting muscle excitation is calculated as $u^{muscle} = \max(u^{CPG}, sens\_sens \cdot u^{reflex})$ for each muscle.

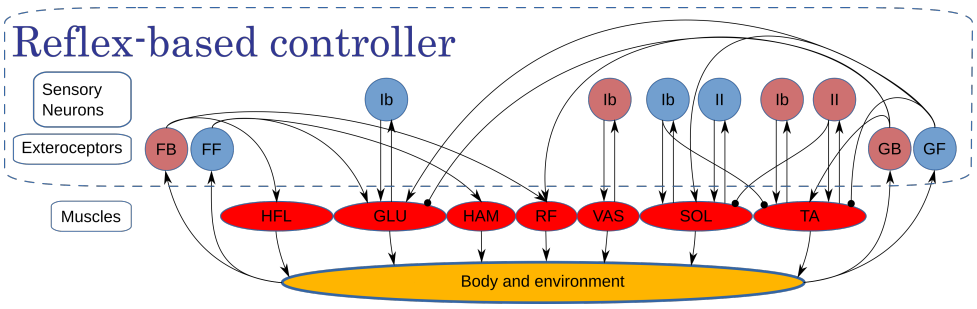


Figure 5.32: Details of reflex balance controller. Lines ending with arrow and circle stand for excitatory and inhibitory connections respectively.

Current implementation of the controller is able to balance for about 3 seconds with additional time standing on toes while falling forward (see **OpenSim reflex balance**[14]). Indeed, standing on toes is a final measure of keeping balance before making a step forward. The latter behavior of a single balancing step is not implemented in this controller. Nevertheless, the reflexive balance controller is capable of its primary function—keeping balance before the start of gait—for which a couple of seconds is enough.

Fig. 5.33 shows signals in balance controller during 3 seconds of equilibrium control. After that, the model falls and the simulation is terminated. The first row of plots contain joint angle values, the each following row is dedicated to one of controlled muscles. In this simulation, the control of right and left leg is the same, thus the plots of right leg (blue line) are situated below the ones for left leg (orange line).

Most controlled muscles have similar constant input, except ankle muscle groups, SOL and TA. These groups are definitive in keeping body balance and fail by the end of simulation due to non-compensated force positive feedback loops. A possible ways to improve this behavior may be larger control of trunk and adding modulating connections to positive feedback loops.

[14] https://mega.nz/#F!sx1hyBIb!WmACHs6ZVGNt08MzmzpZfA

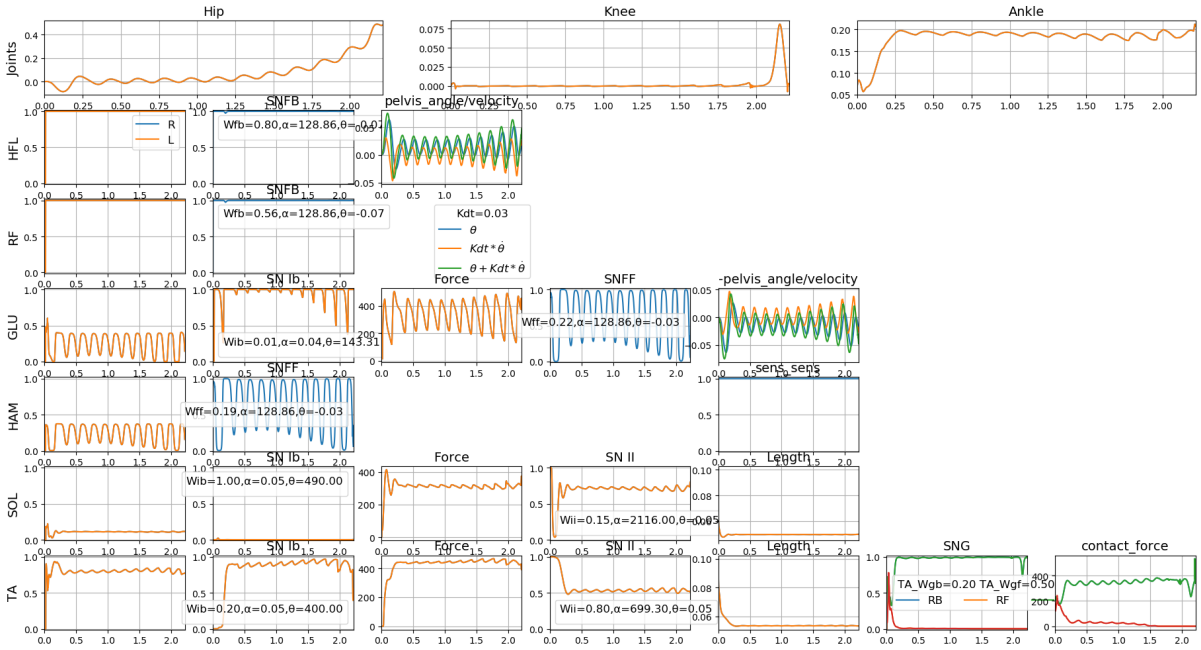


Figure 5.33: Signals from the reflex-based balance controller during equilibrium control simulation.

5.6 Conclusion

This final chapter describes variants of controller architectures for musculoskeletal gait and results of simulations using different controllers. At start, we introduce the chain of neuro-musculoskeletal gait simulation with the example of a single joint control. Then, we describe three CPG controller compositions, each with different control target. And, finally, we demonstrate the work of separate balance controller, based on reflex loops, intended to supplement the CPG controller during gait and stops.

Conclusion

This thesis work has proposed a neuro-musculoskeletal gait simulation platform, featuring forward-time dynamics in OpenSim simulator, simulation of muscle-actuated lower human body, controlled by original model of CPG. Our platform is able to simulate normal human gait as well as gait with a sudden stop and recovery. The controller model takes into account the spinal structures involved in rhythmic movements, such as central pattern generators, their connections to higher-level structures that are affected by Parkinson's disease (basal ganglia, dopamine neurons), and known mechanisms associated with this disease. The model is based on biological signals (movement, activity of the brain and muscles) and is confirmed by simulating of the gait in the model of the musculoskeletal system.

In the Chapter 1, we have introduced the motivation and the final goal of the work: simulation of neurodegenerative diseases' effect on gait. The gait deficiencies are defined from two sides: by their effect on the gait and by their origin, as in Parkinson's disease. The details on the origin of motor control further describe the effects of neural impairments on gait. The existing models of the influence of Parkinson's disease on human gait were analyzed.

In the Chapter 2, the concept of central pattern generator has been described, with their biological evidence, current models, and their application in bio-modeling and control, including robots. We have explained the features of our model of CPG as well as its functioning and composition. And, finally, we have described other modules, relevant to gait and balance control.

In the Chapter 3, we have described the two used musculoskeletal simulators, their features and required modifications to adapt them for our needs of bio-inspired CPG control in closed loop with realistic afferent feedback. First simulator uses simple musculoskeletal model in MATLAB with 16 muscles and 7 body-parts. It is suitable for initial development of the controller, but is limited in extensions and anatomical correctness. Second simulator uses OpenSim physical environment and model that is balanced between anatomical correctness and speed of simulation while being easily extendable. The chapter is finalized by the details on implementation of the simulation platform.

In the Chapter 4 the evaluation and analysis of the resulting simulated gaits has been evaluated. Simulated gaits are to be decomposed and compared to real human gait data, that can be measured. Both real and simulated gait are considered by the phases of the gait cycle and compared with cross-correlation algorithm. The resulting similarity value is used in optimization of gait controller parameters.

The Chapter 5 has combined results of previous chapters to provide simulations of human gaits with CPG controller in different architectures. Each architecture has been described before presenting and analyzing the simulation results. Architectures vary in control target for a CPG: a joint, a muscle group, or a gait phase. The chapter finishes with results in solely balance control of a model.

Discussion

We have presented a new way to simulate musculoskeletal gait of human using central pattern generators. For the first time, a model of the entire motor coordination chain was built, from the midbrain through spinal neurons to the muscles.

The model of the motor coordination is modular, consisting of three parts: upper controller, spinal nervous network, and musculoskeletal model. Each part is interchangeable, requiring several control signals for the other parts.The upper controller is expected to provide three signals to the spinal network: α_{MLR} to control the speed of gait; θ_{MLR} for start/stop commands; and $sens\_sens$ to inhibit spinal reflexes during gait. Depending on the structure of the upper controller, it may utilize feedback signals from the other parts. In this thesis work, the upper controller is presented as a simple threshold function with no feedback, able to initiate gait at the start of simulation, inhibiting the reflexes, as well as simulating a sudden stop during gait, imitating a Freezing of Gait symptom of Parkinson's disease.

The spinal nervous network, presented in this work as central pattern generators and a separate reflexive balance controller, requires the former motor commands from the upper controller and provides excitation signals for each muscle of the musculoskeletal model. From which the spinal network expects muscular afferent feedback of type-*Ia*, *Ib*, and *II*. The feedback to the spinal network is implemented as proprioceptors of human muscles, cutaneous receptors, and equilibrium signal.

The spinal controller is a set of interconnected CPGs, based on mesoscopic bio-inspired model of neuron. The size of CPG set depends on the control target. This thesis work presents three CPG architectures corresponding to three control targets.The joint-targeted controller originates in the robotic control, where a CPG is dedicated to control a single joint or a degree of freedom of a joint. This architecture is suitable for simple musculoskeletal models with monoarticular muscles, so the CPG's muscles wouldn't interfere with in different joints.The muscle-targeted controller is the most advanced one, dedicating a CPG to each antagonistic pair of muscles. This architecture allows to tune each muscle pair's activation phase as well as the other CPG parameters. But it results in a large amount of parameters, making controller optimization process overly complicated.

And lastly, the phase-targeted controller simplifies gait control, grouping controlled muscles for CPG by their activation phase during gait. For this architecture, we need just two CPG per leg, while limiting ourselves with one gait type. The less amount of CPGs also means easier parameter tuning, even allowing manual parameter variation from CPGs function and gait timings. This controller may be expanded with muscle synergy approach, discussed further.

The model of musculoskeletal system receives muscle excitation, performs the movement, and returns the mentioned feedback signals to the controllers. Performed movement is realizes as forward-time dynamics calculations during each time step and returning feedback in-between.

The final musculoskeletal simulator in this work is OpenSim with implemented anatomical model of muscle-actuated lower human body and torque-actuated arms. This model is easily expandable, allowing to add muscles for upper body and arms.Our simulation platform for gait is a notable step forward in reaching the goal of simulating abnormal gaits that originate in the midbrain. These gaits include neurodegenerative diseases, such as Parkinson's. The ability to simulate various gaits opens a possibility of medical application of our simulation platform, both in research and in diagnosis and treatment planning. Research may benefit from gait analysis and simulation of unobtainable or invasive data and treatment

specialists could use our platform to predict interventions in musculoskeletal and nervous systems.

As a result, our platform is able to produce stable gait in sagittal plane using musculoskeletal model. Simulation results show that changing intrinsic parameters of controller produce different types of gait with a possibility to change parameters of gait through a signal from upper controller, simulating a decision-making function of a midbrain. Following this way, we assume that the neuro-musculoskeletal simulator presented here will be able to simulate impacts of PD disorders on human walking like observed in the medical studies.

Muscle synergy hypothesis and its application to phase-targeted CPG

Throughout the description of the phase-targeted CPG controller, we have mentioned the muscle synergy hypothesis. This is a motor controlling manner, aimed to resolve the problem of muscle control redundancy. The problem lays in the fact that the amount of muscles in the human body greatly exceed the number of degrees of freedom. And the existence of specialized spinal nervous networks that transform the motor control, descending from the brain, into muscle excitations.

Unlike the CPG hypothesis, the muscle synergy suggests the function of spinal networks as burst generators of specific motor primitives that produce desired movements. These motor primitives are usually obtained with non-negative matrix factorization algorithm from EMG records. Each of the several primitives is suggested to be an output from neurons of a spine segment. All of the primitives, weighted and combined result in a specific movement.

In a review [112], Degallier et al. have described the features of muscle synergy control, with their biological evidence. In our opinion, some of these features are compatible with our CPG controller. Indeed, a feature of discrete movement occurring often in a specific phase window of oscillations, is reproducible and even bounding for our CPG during the change of its oscillation mode from rhythmic to plateau. Although, this feature was not used in current thesis work. The feature of discrete movements suppressing rhythmic ones is similar. There is a single CPG for both modes, switching between them on an upper command.

The similarities between muscle synergy modules and our model of CPG controller suggest a possibility to combine both approaches to profit from their strong sides. For the synergy, it is an understandable anatomical location and defined function of each primitive. As well as a definitive way to estimate each primitive through non-negative matrix factorization. CPG-based approach offers continuous pattern generation, in-command interrupted with discrete movements. As well as flexible architecture with tuneable parameters.

Indeed, muscle primitives naturally grew from the phase-targeted CPG controller approach, when each CPG required different weights for each muscle. All is left is to connect each muscle to more than one defined CPG with weights, defined from non-negative matrix factorization. In other words, we propose to use CPG to generate motor primitives. With our controller, we assume that four primitives, two unique and other two are phase-shifted by 180 degrees, would be enough to control gait. And this amount is in order with the other synergy approaches.

Future work

This thesis does not finalizes the work and leaves several directions for potential future research. But first, we would like to discuss the places left to improve in current work.

First of all, the supportive harness was added because the model didn't have the means to balance itself. Later, a reflexive balance controller was added, assumed to help the model with upward position. While it is able to keep balance in sagittal plane for several seconds at the start of simulation, it is not sufficient for real equilibrium. The bipedal gait implies active role of brain structures, e.g. cerebellum, in balance control.

The second place left to improve is a primitive upper controller, implemented as a threshold function that starts and stops the gait. While the gait initiation from standing position on timer is sufficient, the same way of proper halting the gait is too basic. Meaning, that it does not involve the position of legs during the stop or any other kind of feedback, which leads to unnatural leg position on stop. In Chapter 1, we have described models of decision-making systems of brain, a couple of which are possible to adapt for our cause.

Short-term perspectives

There are also research directions, left out from the current work to avoid too much distraction from the main goal. First, we put the several simpler ones. We suggest to try out controlling α_{MLR} signal of CPG, assumed to vary the stride length and, therefore, the speed of gait. And to compare resulting dependencies of step length on gait speed with humans. Then, we could include additional gait characteristics into optimization target function, like hip height, zero-moment point from robotics, or augment our correlation procedure to account for several different types of gait in one simulation. Currently, the oscillation frequency of CPG is chosen arbitrarily in the range of the average human walk velocity, but we assume a way to calculate the inner frequency for CPG from simplified inverted pendulum model of body model. Another research direction would be to follow biological evidence and inhibit reflexes not with a single sensory sensitivity signal, but condition it at each for reflex arcs and gait cycle phase [44].

Long-term perspectives

The more complicated research directions include releasing the musculoskeletal model movements from sagittal plane only, making control three-dimensional and return removed muscles. This greatly increases control difficulty and computational complexity of simulation.

Further work includes more computational experiments with different gaits including altered ones. Chapter 1 suggests classification of simulated gaits by their parameters, although it would require a large amount of work on gait classifier and their recognition. After that, such classifier would allow to choose the desired type of altered gait we want to simulate. The simulated symptoms of Parkinson's disease include: Freezing of Gait, tremor, muscular rigidity, slow and imprecise movement. Though, further work would be aimed at simulation of altered walking gaits, based on patients' data. The other possible abnormal gaits are limping, ataxic gait, myopathy, cerebral palsy-altered gaits, etc.

Another research may be directed at asymmetry of human gait, such as existence of "pushing leg" phenomena and irregularity of walking periods. Further aiming on asymmetry may include experiments

on asymmetric CPG models [44], where upper controller activates only flexion half-center during swing and extension one is activated by loading sensors (type-Ib) at the start of stance phase. As well as coupling RG layer with biped by afferent feedback (type-II) as i_{inj} or experimenting on 3 rhythmic neurons together, like Matsuoka did (see the model in Chapter 2).

Bibliography

[1] T. Hanakawa, Y. Katsumi, H. Fukuyama, M. Honda, T. Hayashi, J. Kimura, and H. Shibasaki, "Mechanisms underlying gait disturbance in Parkinson's disease: a single photon emission computed tomography study," *Brain: A Journal of Neurology*, vol. 122 (Pt 7), pp. 1271–1282, July 1999.

[2] T. G. Brown, "On the nature of the fundamental activity of the nervous centres; together with an analysis of the conditioning of rhythmic activity in progression, and a theory of the evolution of function in the nervous system," *The Journal of Physiology*, vol. 48, pp. 18–46, Mar. 1914.

[3] Ton van den Bogert, "Musculoskeletal Model for Simulation of Walking — Human Motion and Control Laboratory," 2011.

[4] A. Rajagopal, C. L. Dembia, M. S. DeMers, D. D. Delp, J. L. Hicks, and S. L. Delp, "Full-Body Musculoskeletal Model for Muscle-Driven Simulation of Human Gait," *IEEE transactions on bio-medical engineering*, vol. 63, no. 10, pp. 2068–2079, 2016.

[5] S. L. Delp, F. C. Anderson, A. S. Arnold, P. Loan, A. Habib, C. T. John, E. Guendelman, and D. G. Thelen, "OpenSim: open-source software to create and analyze dynamic simulations of movement," *IEEE transactions on bio-medical engineering*, vol. 54, pp. 1940–1950, Nov. 2007.

[6] J. Nassour, P. Henaff, F. B. Ouezdou, and G. Cheng, "Bipedal Locomotion Control with Rhythmic Neural Circuits," in *International Workshop on Bio-Inspired Robots*, (nantes, France), p. xx, Apr. 2011.

[7] Y. Asano, T. Shirai, T. Kozuki, Y. Motegi, Y. Nakanishi, K. Okada, and M. Inaba, "Motion generation of redundant musculoskeletal humanoid based on robot-model error compensation by muscle load sharing and interactive control device," in *2013 13th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 336–341, Oct. 2013.

[8] S. Kurumaya, K. Suzumori, H. Nabae, and S. Wakimoto, "Musculoskeletal lower-limb robot driven by multifilament muscles," *ROBOMECH Journal*, vol. 3, p. 18, Sept. 2016.

[9] G. Palli, G. Borghesan, and C. Melchiorri, "Modeling, Identification, and Control of Tendon-Based Actuation Systems," *IEEE Transactions on Robotics*, vol. 28, pp. 277–290, Apr. 2012.

[10] N. Giladi, F. B. Horak, and J. M. Hausdorff, "Classification of gait disturbances: distinguishing between continuous and episodic changes," *Movement disorders : official journal of the Movement Disorder Society*, vol. 28, Sept. 2013.

[11] F. Chantraine, P. Filipetti, C. Schreiber, A. Remacle, E. Kolanowski, and F. Moissenet, "Proposition of a Classification of Adult Patients with Hemiparesis in Chronic Phase," *PLOS ONE*, vol. 11, p. e0156726, June 2016.

[12] J.-P. Azulay, "Locomotion : physiologie, méthodes d'analyse et classification des principaux troubles," *ResearchGate*, 2015.

[13] J. G. Nutt, F. B. Horak, and B. R. Bloem, "Milestones in gait, balance, and falling," *Movement Disorders: Official Journal of the Movement Disorder Society*, vol. 26, pp. 1166–1174, May 2011.

[14] O. Kiehn and K. Dougherty, "Locomotion: Circuits and Physiology," in *Neuroscience in the 21st Century* (D. W. Pfaff, ed.), pp. 1209–1236, New York, NY: Springer New York, 2013.

[15] J. G. Nutt, B. R. Bloem, N. Giladi, M. Hallett, F. B. Horak, and A. Nieuwboer, "Freezing of gait: moving forward on a mysterious clinical phenomenon," *The Lancet. Neurology*, vol. 10, pp. 734–744, Aug. 2011.

[16] V. S. Chakravarthy, D. Joseph, and R. S. Bapi, "What do the basal ganglia do? A modeling perspective," *Biological Cybernetics*, vol. 103, pp. 237–253, Sept. 2010.

[17] N. Mallet, B. Micklem, P. Henny, M. Brown, C. Williams, J. P. Bolam, K. Nakamura, and P. Magill, "Dichotomous Organization of the External Globus Pallidus," *Neuron*, vol. 74, pp. 1075–1086, June 2012.

[18] T. J. Prescott, F. M. Montes González, K. Gurney, M. D. Humphries, and P. Redgrave, "A robot model of the basal ganglia: behavior and intrinsic processing," *Neural Networks: The Official Journal of the International Neural Network Society*, vol. 19, pp. 31–61, Jan. 2006.

[19] J. L. Contreras-Vidal and G. E. Stelmach, "A neural model of basal ganglia-thalamocortical relations in normal and parkinsonian movement," *Biological Cybernetics*, vol. 73, pp. 467–476, Oct. 1995.

[20] H. L. Teulings, J. L. Contreras-Vidal, G. E. Stelmach, and C. H. Adler, "Parkinsonism reduces coordination of fingers, wrist, and arm in fine motor control," *Experimental Neurology*, vol. 146, pp. 159–170, July 1997.

[21] J. A. Obeso, M. C. Rodríguez-Oroz, B. Benitez-Temino, F. J. Blesa, J. Guridi, C. Marin, and M. Rodriguez, "Functional organization of the basal ganglia: therapeutic implications for Parkinson's disease," *Movement Disorders: Official Journal of the Movement Disorder Society*, vol. 23 Suppl 3, pp. S548–559, 2008.

[22] A. Pavlides, S. J. Hogan, and R. Bogacz, "Computational Models Describing Possible Mechanisms for Generation of Excessive Beta Oscillations in Parkinson's Disease," *PLOS Comput Biol*, vol. 11, p. e1004609, Dec. 2015.

[23] A. Dovzhenok and L. L. Rubchinsky, "On the Origin of Tremor in Parkinson's Disease," *PLOS ONE*, vol. 7, p. e41598, July 2012.

[24] L. Hirsch, N. Jette, A. Frolkis, T. Steeves, and T. Pringsheim, "The Incidence of Parkinson's Disease: A Systematic Review and Meta-Analysis," *Neuroepidemiology*, vol. 46, pp. 292–300, Apr. 2016. #01.

[25] W. Dauer and S. Przedborski, "Parkinson's Disease: Mechanisms and Models," *Neuron*, vol. 39, pp. 889–909, Sept. 2003.

[26] GBD 2015 Neurological Disorders Collaborator Group, "Global, regional, and national burden of neurological disorders during 1990-2015: a systematic analysis for the Global Burden of Disease Study 2015," *The Lancet. Neurology*, vol. 16, no. 11, pp. 877–897, 2017.

[27] L. Bertram and R. E. Tanzi, "The genetic epidemiology of neurodegenerative disease," *Journal of Clinical Investigation*, vol. 115, pp. 1449–1457, June 2005.

[28] A. H. Snijders, I. Leunissen, M. Bakker, S. Overeem, R. C. Helmich, B. R. Bloem, and I. Toni, "Gait-related cerebral alterations in patients with Parkinson's disease with freezing of gait," *Brain: A Journal of Neurology*, vol. 134, pp. 59–72, Jan. 2011.

[29] S. Donovan, C. Lim, N. Diaz, N. Browner, P. Rose, L. R. Sudarsky, D. Tarsy, S. Fahn, and D. K. Simon, "Laserlight cues for gait freezing in Parkinson's disease: an open-label study," *Parkinsonism & Related Disorders*, vol. 17, pp. 240–245, May 2011.

[30] A. Delval, A. H. Snijders, V. Weerdesteyn, J. E. Duysens, L. Defebvre, N. Giladi, and B. R. Bloem, "Objective detection of subtle freezing of gait episodes in Parkinson's disease," *Movement Disorders: Official Journal of the Movement Disorder Society*, vol. 25, pp. 1684–1693, Aug. 2010.

[31] P. Arias and J. Cudeiro, "Effect of Rhythmic Auditory Stimulation on Gait in Parkinsonian Patients with and without Freezing of Gait," *PLOS ONE*, vol. 5, p. e9675, Mar. 2010.

[32] S. Rahman, H. J. Griffin, N. P. Quinn, and M. Jahanshahi, "The factors that induce or overcome freezing of gait in Parkinson's disease," *Behavioural Neurology*, vol. 19, no. 3, pp. 127–136, 2008.

[33] A. Nieuwboer, "Cueing for freezing of gait in patients with Parkinson's disease: A rehabilitation perspective," *Movement Disorders*, vol. 23, pp. S475–S481, Jan. 2008.

[34] G. Gelders, V. Baekelandt, and A. Van der Perren, "Linking Neuroinflammation and Neurodegeneration in Parkinson's Disease," *Journal of Immunology Research*, vol. 2018, Apr. 2018.

[35] A. Gupta, P. P. Balasubramani, and V. S. Chakravarthy, "Computational model of precision grip in Parkinson's disease: a utility based approach," *Frontiers in Computational Neuroscience*, vol. 7, p. 172, 2013.

[36] V. Muralidharan, P. P. Balasubramani, V. S. Chakravarthy, S. J. G. Lewis, and A. A. Moustafa, "A computational model of altered gait patterns in parkinson's disease patients negotiating narrow doorways," *Frontiers in Computational Neuroscience*, vol. 7, p. 190, 2014.

[37] Y. Tachibana, H. Iwamuro, H. Kita, M. Takada, and A. Nambu, "Subthalamo -pallidal interactions underlying parkinsonian neuronal oscillations in the primate basal ganglia," *European Journal of Neuroscience*, vol. 34, pp. 1470–1484, Nov. 2011.

[38] A. C. Marreiros, H. Cagnan, R. J. Moran, K. J. Friston, and P. Brown, "Basal ganglia–cortical interactions in Parkinsonian patients," *NeuroImage*, vol. 66, pp. 301–310, Feb. 2013.

[39] P. A. Guertin, "The mammalian central pattern generator for locomotion," *Brain Research Reviews*, vol. 62, pp. 45–56, Dec. 2009.

[40] P. F. Rowat and A. I. Selverston, "Modeling the gastric mill central pattern generator of the lobster with a relaxation-oscillator network," *Journal of Neurophysiology*, vol. 70, pp. 1030–1053, Sept. 1993.

[41] I. A. Rybak, N. A. Shevtsova, M. Lafreniere-Roula, and D. A. McCrea, "Modelling spinal circuitry involved in locomotor pattern generation: insights from deletions during fictive locomotion," *The Journal of Physiology*, vol. 577, pp. 617–639, Dec. 2006.

[42] P. A. Guertin, "Central Pattern Generator for Locomotion: Anatomical, Physiological, and Pathophysiological Considerations," *Frontiers in Neurology*, vol. 3, 2013.

[43] M. MacKay-Lyons, "Central pattern generation of locomotion: a review of the evidence," *Physical Therapy*, vol. 82, pp. 69–83, Jan. 2002.

[44] J. Duysens and A. Forner-Cordero, "A controller perspective on biological gait control: Reflexes and central pattern generators," *Annual Reviews in Control*, p. S1367578818302177, Apr. 2019.

[45] S. Yakovenko, "A hierarchical perspective on rhythm generation for locomotor control," *Progress in Brain Research*, vol. 188, pp. 151–166, 2011.

[46] D. Zhang, P. Poignet, A. P. L. Bo, and W. T. Ang, "Exploring peripheral mechanism of tremor on neuromusculoskeletal model: a general simulation study," *IEEE transactions on bio-medical engineering*, vol. 56, pp. 2359–2369, Oct. 2009.

[47] K. Jansen, F. De Groote, W. Aerts, J. De Schutter, J. Duysens, and I. Jonkers, "Altering length and velocity feedback during a neuro-musculoskeletal simulation of normal gait contributes to hemiparetic gait characteristics," *Journal of NeuroEngineering and Rehabilitation*, vol. 11, p. 78, Apr. 2014.

[48] A. Murai, K. Yamane, and Y. Nakamura, "Modeling and identification of human neuromusculoskeletal network based on biomechanical property of muscle," *Conference proceedings: ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference*, vol. 2008, pp. 3706–3709, 2008.

[49] S. Aoi, T. Ohashi, R. Bamba, S. Fujiki, D. Tamura, T. Funato, K. Senda, Y. Ivanenko, and K. Tsuchiya, "Neuromusculoskeletal model that walks and runs across a speed range with a few motor control parameter changes based on the muscle synergy hypothesis," *Scientific Reports*, vol. 9, p. 369, Jan. 2019.

[50] F. De Groote, I. Jonkers, and J. Duysens, "Task constraints and minimization of muscle effort result in a small number of muscle synergies during gait," *Frontiers in Computational Neuroscience*, vol. 8, 2014.

[51] L. H. Ting, H. J. Chiel, R. D. Trumbower, J. L. Allen, J. L. McKay, M. E. Hackney, and T. M. Kesar, "Neuromechanical principles underlying movement modularity and their implications for rehabilitation," *Neuron*, vol. 86, pp. 38–54, Apr. 2015.

[52] L. H. Ting and J. L. McKay, "Neuromechanics of muscle synergies for posture and movement," *Current Opinion in Neurobiology*, vol. 17, pp. 622–628, Dec. 2007.

[53] E. Amrollah and P. Henaff, "On the Role of Sensory Feedbacks in Rowat–Selverston CPG to Improve Robot Legged Locomotion," *Frontiers in Neurorobotics*, vol. 4, Dec. 2010.

[54] S. Rossignol, R. Dubuc, and J.-P. Gossard, "Dynamic sensorimotor interactions in locomotion," *Physiological Reviews*, vol. 86, pp. 89–154, Jan. 2006.

[55] L. M. Jordan, J. Liu, P. B. Hedlund, T. Akay, and K. G. Pearson, "Descending command systems for the initiation of locomotion in mammals," *Brain Research Reviews*, vol. 57, pp. 183–191, Jan. 2008.

[56] R. J. Peterka, "Sensory integration for human balance control," in *Handbook of Clinical Neurology*, vol. 159, pp. 27–42, Elsevier, 2018.

[57] M. Goulding, "Circuits controlling vertebrate locomotion: moving in a new direction," *Nature Reviews Neuroscience*, vol. 10, pp. 507–518, July 2009.

[58] I. A. Rybak, K. J. Dougherty, and N. A. Shevtsova, "Organization of the Mammalian Locomotor CPG: Review of Computational Model and Circuit Architectures Based on Genetically Identified Spinal Interneurons," *eNeuro*, vol. 2, pp. ENEURO.0069–15.2015, Sept. 2015.

[59] J. Ausborn, N. A. Shevtsova, V. Caggiano, S. M. Danner, and I. A. Rybak, "Computational modeling of brainstem circuits controlling locomotor frequency and gait," *eLife*, vol. 8, p. e43587, Jan. 2019.

[60] D. G. Ivashko, B. I. Prilutsky, S. N. Markin, J. K. Chapin, and I. A. Rybak, "Modeling the spinal cord neural circuitry controlling cat hindlimb movement during locomotion," *Neurocomputing*, vol. 52–54, pp. 621–629, June 2003.

[61] N. S. Szczecinski, A. J. Hunt, and R. D. Quinn, "Design process and tools for dynamic neuromechanical models and robot controllers," *Biological Cybernetics*, vol. 111, pp. 105–127, Feb. 2017.

[62] M. Sekerli and R. J. Butera, "Oscillations in a Simple Neuromechanical System: Underlying Mechanisms," *Journal of Computational Neuroscience*, vol. 19, pp. 181–197, Oct. 2005.

[63] A. J. Ijspeert, "Central pattern generators for locomotion control in animals and robots: A review," *Neural Networks*, vol. 21, pp. 642–653, May 2008. #15.

[64] P. F. Rowat and A. I. Selverston, "Learning algorithms for oscillatory networks with gap junctions and membrane currents," *Network: Computation in Neural Systems*, vol. 2, pp. 17–41, Jan. 1991.

[65] E. Marder and D. Bucher, "Central pattern generators and the control of rhythmic movements," *Current Biology*, vol. 11, pp. R986–R996, Nov. 2001.

[66] K. Matsuoka, "Sustained oscillations generated by mutually inhibiting neurons with adaptation," *Biological Cybernetics*, vol. 52, pp. 367–376, Oct. 1985.

[67] G. Taga, "A model of the neuro-musculo-skeletal system for human locomotion," *Biological Cybernetics*, vol. 73, pp. 97–111, July 1995.

[68] N. Ogihara and N. Yamazaki, "Generation of human bipedal locomotion by a bio-mimetic neuro-musculo-skeletal model," *Biological Cybernetics*, vol. 84, pp. 1–11, Jan. 2001.

[69] K. Hase, K. Miyashita, S. Ok, and Y. Arakawa, "Human gait simulation with a neuromusculoskeletal model and evolutionary computation," *The Journal of Visualization and Computer Animation*, vol. 14, pp. 73–92, May 2003.

[70] B. W. Verdaasdonk, H. F. J. M. Koopman, and F. C. T. van der Helm, "Energy efficient walking with central pattern generators: from passive dynamic walking to biologically inspired control," *Biological Cybernetics*, vol. 101, pp. 49–61, July 2009.

[71] Y. Kim, Y. Tagawa, G. Obinata, and K. Hase, "Robust control of CPG-based 3d neuromusculoskeletal walking model," *Biological Cybernetics*, vol. 105, pp. 269–282, Oct. 2011.

[72] M. Jouaiti, L. Caron, and P. Hénaff, "Hebbian Plasticity in CPG Controllers Facilitates Self-Synchronization for Human-Robot Handshaking," *Frontiers in Neurorobotics*, vol. 12, p. 29, 2018.

[73] Balth. van der Pol Jun. D.Sc, "LXXXVIII. On "relaxation-oscillations"," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 7, no. 2:11, pp. 978–992, 1926.

[74] J. Nassour, T. D. Hoa, P. Atoofi, and F. Hamker, "Concrete Action Representation Model: from Neuroscience to Robotics," *IEEE Transactions on Cognitive and Developmental Systems*, pp. 1–1, 2019.

[75] L. Righetti and A. J. Ijspeert, "Programmable central pattern generators: an application to biped locomotion control," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pp. 1585–1590, May 2006. #25.

[76] A. J. Ijspeert, A. Crespi, D. Ryczko, and J.-M. Cabelguen, "From swimming to walking with a salamander robot driven by a spinal cord model," *Science (New York, N.Y.)*, vol. 315, pp. 1416–1420, Mar. 2007.

[77] S. Jo and S. G. Massaquoi, "A model of cerebrocerebello-spinomuscular interaction in the sagittal control of human walking," *Biological Cybernetics*, vol. 96, pp. 279–307, Mar. 2007.

[78] K. B. Arikan and B. Irfanoglu, "A Test Bench to Study Bioinspired Control for Robot Walking," *Journal of Control Engineering and Applied Informatics*, vol. 13, pp. 76–80–80, June 2011.

[79] A. Melnyk and P. Henaff, "Bio-inspired plastic controller for a robot arm to shake hand with human," in *2016 IEEE 36th International Conference on Electronics and Nanotechnology (ELNANO)*, pp. 163–168, Apr. 2016.

[80] T. Geng, B. Porr, and F. Worgotter, "Coupling of Neural Computation with Physical Computation for Stable Dynamic Biped Walking Control," p. 43, 2006.

[81] P. Manoonpong, T. Geng, T. Kulvicius, B. Porr, and F. Wörgötter, "Adaptive, Fast Walking in a Biped Robot under Neuronal Control and Learning," *PLOS Computational Biology*, vol. 3, p. e134, July 2007. #20.

[82] J. Nassour, P. Hénaff, F. Benouezdou, and G. Cheng, "Multi-layered multi-pattern CPG for adaptive locomotion of humanoid robots," *Biological Cybernetics*, vol. 108, pp. 291–303, June 2014.

[83] N. Van der Noot, A. J. Ijspeert, and R. Ronsse, "Neuromuscular model achieving speed control and steering with a 3d bipedal walker," *Autonomous Robots*, vol. 43, pp. 1537–1554, Aug. 2019.

[84] J. Yu, M. Tan, J. Chen, and J. Zhang, "A survey on CPG-inspired control models and system implementation," *IEEE transactions on neural networks and learning systems*, vol. 25, pp. 441–456, Mar. 2014.

[85] C. Paul, M. Bellotti, S. Jezernik, and A. Curt, "Development of a human neuro-musculo-skeletal model for investigation of spinal cord injury," *Biological Cybernetics*, vol. 93, pp. 153–170, Sept. 2005.

[86] A. Melnyk, P. Hénaff, and A. Popov, "Analysis of a handshake between humans using wavelet transforms," in *2015 IEEE 35th International Conference on Electronics and Nanotechnology (ELNANO)*, pp. 397–401, Apr. 2015.

[87] D. A. McCrea and I. A. Rybak, "Organization of mammalian locomotor rhythm and pattern generation," *Brain Research Reviews*, vol. 57, pp. 134–146, Jan. 2008.

[88] V. Dietz, "Do human bipeds use quadrupedal coordination?," *Trends in Neurosciences*, vol. 25, pp. 462–467, Sept. 2002.

[89] S. Song and H. Geyer, "A neural circuitry that emphasizes spinal feedback generates diverse behaviours of human locomotion," *The Journal of Physiology*, vol. 593, pp. 3493–3511, Aug. 2015.

[90] A. Shachykov, P. Henaff, A. Popov, and A. Shulyak, "CPG-based circuitry for controlling musculoskeletal model of human locomotor system," in *2017 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pp. 1–4, Oct. 2017.

[91] A. Pachi and T. Ji, "Frequency and velocity of people walking," *ResearchGate*, Feb. 2005.

[92] K. R. Saul, X. Hu, C. M. Goehler, M. E. Vidt, M. Daly, A. Velisar, and W. M. Murray, "Benchmarking of dynamic simulation predictions in two software platforms using an upper limb musculoskeletal model," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 18, no. 13, pp. 1445–1458, 2015.

[93] E. M. Arnold, S. R. Hamner, A. Seth, M. Millard, and S. L. Delp, "How muscle fiber lengths and velocities affect muscle force generation as humans walk and run at different speeds," *The Journal of Experimental Biology*, vol. 216, pp. 2150–2160, June 2013.

[94] O. Röhrle, J. B. Davidson, and A. J. Pullan, "A Physiologically Based, Multi-Scale Model of Skeletal Muscle Structure and Function," *Frontiers in Physiology*, vol. 3, 2012.

[95] A. V. Hill, "The Heat of Shortening and the Dynamic Constants of Muscle," *Proceedings of the Royal Society of London B: Biological Sciences*, vol. 126, pp. 136–195, Oct. 1938.

[96] D. G. Thelen, "Adjustment of muscle mechanics model parameters to simulate dynamic contractions in older adults," *Journal of Biomechanical Engineering*, vol. 125, pp. 70–77, Feb. 2003.

[97] M. Ackermann and Antonie J. van den Bogert, "Optimality principles for model-based prediction of human gait," *Journal of Biomechanics*, vol. 43, pp. 1055–1060, Apr. 2010.

[98] F. E. Zajac, "Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control," *Critical Reviews in Biomedical Engineering*, vol. 17, no. 4, pp. 359–411, 1989.

[99] M. Millard, T. Uchida, A. Seth, and S. L. Delp, "Flexing computational muscle: modeling and simulation of musculotendon dynamics," *Journal of Biomechanical Engineering*, vol. 135, p. 021005, Feb. 2013.

[100] J. P. Hunter, P. Ashby, and A. E. Lang, "Afferents contributing to the exaggerated long latency reflex response to electrical stimulation in Parkinson's disease.," *Journal of Neurology, Neurosurgery, and Psychiatry*, vol. 51, pp. 1405–1410, Nov. 1988.

[101] The MathWorks Inc., "MATLAB," 2017.

[102] A. Shachykov, P. Henaff, A. Popov, and A. Shulyak, "Neuro-musculoskeletal simulator of human rhythmic movements," in *2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)*, pp. 278–283, May 2017.

[103] C. Mummolo, L. Mangialardi, and J. H. Kim, "Quantifying dynamic characteristics of human walking for comprehensive gait cycle," *Journal of Biomechanical Engineering*, vol. 135, p. 91006, Sept. 2013.

[104] C. Beyaert, R. Vasa, and G. E. Frykberg, "Gait post-stroke: Pathophysiology and rehabilitation strategies," *Neurophysiologie Clinique = Clinical Neurophysiology*, vol. 45, pp. 335–355, Nov. 2015.

[105] J.-P. Azulay, S. Cantiniaux, F. Vacherot, M. Vaugoyeau, and C. Assaiante, "Locomotion: physiologie, méthodes d'analyse et classification des principaux troubles," 2009.

[106] P. Knysh and Y. Korkolis, "Blackbox: A procedure for parallel optimization of expensive black-box functions," *arXiv:1605.00998 [cs, math]*, May 2016. arXiv: 1605.00998.

[107] J. E. A. Bertram and A. Ruina, "Multiple Walking Speed–frequency Relations are Predicted by Constrained Optimization," *Journal of Theoretical Biology*, vol. 209, pp. 445–453, Apr. 2001.

[108] A. P. Shulyak and A. D. Shachykov, "About the impact of informative features selection in the mutually orthogonal decompositions of biomedical signals for their recognition," in *2016 IEEE 36th International Conference on Electronics and Nanotechnology (ELNANO)*, pp. 228–231, Apr. 2016.

[109] H. Serhan and P. Henaff, "Muscle-Like Compliance in Knee Articulations Improves Biped Robot Walkings," *Recent Advances in Robotic Systems*, 2016.

[110] G. Martino, Y. P. Ivanenko, M. Serrao, A. Ranavolo, A. d'Avella, F. Draicchio, C. Conte, C. Casali, and F. Lacquaniti, "Locomotor patterns in cerebellar ataxia," *Journal of Neurophysiology*, vol. 112, pp. 2810–2821, Dec. 2014.

[111] L. Zhao, L. Zhang, L. Wang, and J. Wang, "Three-dimensional motion of the pelvis during human walking," in *IEEE International Conference Mechatronics and Automation, 2005*, vol. 1, pp. 335–339 Vol. 1, July 2005.

[112] S. Degallier and A. Ijspeert, "Modeling discrete and rhythmic movements through motor primitives: a review," *Biological Cybernetics*, vol. 103, pp. 319–338, Oct. 2010.

My publications

Ukrainian journals

[1] Shachykov, A.D.; Henaff, P.; Shulyak, A.P., “Features of using the characteristics of the form of biomedical signals in their recognition” Bulletin of National Technical University of Ukraine “KPI”. Series Instrument Making, National Technical University of Ukraine “Kyiv Polytechnic Institute”, 2016, 51(1), pp.131-139.

[2] Shulyak, A.P.; Shachykov, A.D., “Criteria and Procedures for Estimating the Informativity and Feature Selection in Biomedical Signals for their Recognition” Bulletin of National Technical University of Ukraine “KPI”. Series of radio engineering. Radio equipment construction, National Technical University of Ukraine “Kyiv Polytechnic Institute”, 2016, 51, p.66.

[3] Shachykov, A.D.; Henaff, P.; Shulyak, A.P., “Neuro-musculoskeletal simulator of the rhythmic movements of human hip joint” Bulletin of National Technical University of Ukraine “KPI”. Series Instrument Making, National Technical University of Ukraine “Kyiv Polytechnic Institute”, 2018, 55(1), pp.118-125.

International conferences

[1] Shulyak, A.P.; Shachykov, A.D., “About the Impact of Informative Features Selection in the Mutually Orthogonal Decompositions of Biomedical Signals for their Recognition”, Electronics and Nanotechnology (ELNANO), 2016 IEEE 36th International Conference, 19-21 April 2016. pp. 228-231.

[2] Shachykov, A.D.; Henaff, P.; Popov, A.A.; Shulyak, A.P., "Neuro-musculoskeletal simulator of human rhythmic movements" 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), 29 May – 2 June 2017, pp. 278-283.

[3] Shachykov, A.D.; Henaff, P.; Popov, A.A.; Shulyak, A.P., "CPG-based circuitry for controlling musculoskeletal model of human locomotor system" 2017 IEEE Biomedical Circuits and Systems Conference (BioCAS), 19 – 21 October 2017, pp. 276-179.

[4] Shachykov A.D., "Comparison of measured and modeled movements of the human lower limbs" Proceedings of the 1st International Scientific and Technical Conference "Actual Problems of Automation and Instrumentation", 7 – 8 December 2017; Kharkiv, pp. 91-92.

[5] Shachykov, A.D.; Henaff, P.; Shulyak, A.P., "Closed-loop Central Pattern Generator Control of Human Gaits in OpenSim Simulator" 2019 IEEE/ International Joint Conference on Neural Networks (IJCNN), July 14 - 19 2019.

National conferences

[1] Shachykov, A.D.; Henaff, P.; Shulyak, A.P., "Modeling of human gait control using CPGs" 2018 Journées Nationales de la Robotique Humanoide (JNRH), Nancy, 14 – 15 June 2018.

Appendix A

GAIT2DE

The GAIT2DE model implements all body dynamics as a single function which needs to be explicitly integrated. In the code fragment below, we use MATLAB's differential equations solver `ode15s()`. Its first argument is integrated function, a lambda-wrapped `gait2de()` in this case. Other arguments are integration time limits and current state of the system.

```
[~, y] = ode15s( @(t,x)gait2de(x,u,M), timegrid([ti-1, ti]), x(:, ti-1) );
x(:,ti) = y(end,:)';
```

here, `y` is intermediate integration result with variable time step, `x` is a state of the system, `u` is input excitation, `M` is external forces applied to the model used for elastic harness, `ti` is the number of current time step, and `timegrid` array holds the global time steps of simulation.

Then, I call `gait2de()` directly to get system state derivatives (`xdot`), ground reaction forces (`grf`), body position (`stick`), and muscle forces (`forces`).

```
[xdot(:,ti), grf(:,ti), stick(:,ti), forces(:,ti)] = gait2de(x(:,ti), u, M);
```

Additionally, output from MN should be limited to $[0; 1]$ as model does not apply such itself.

Table A.1: Parameters of 8-CPG gait controller with GAIT2DE model.

Common parameters					
Name	σ_s	$SNF\alpha$	$SNF\theta$	$SNG\alpha$	$SNG\theta$
Value	50	2.65	0.297	0.0186	131

CPG Hip1 parameters											
Name	V	y	Af	ε	Wc	Wi	$Wsiar$	$PF\alpha$	$PF\theta$	Wfb	Wff
Value	0	4.24	1.54	0.821	0.16	0.124	0.179	11.9	0	0	0
Name	Wgb	Wgf			$MN\alpha$	$MN\theta$	Wpf	Wpe		$Wsia$	$Wsib$
Value	0	0			9.31	3.14	22.6	41.4		15.4	6.78
Name	$Wsii$	Wm	$SNIa\alpha$	$SNIa\theta$	$SNIb\alpha$	$SNIb\theta$	$SNII\alpha$	$SNII\theta$			
Value	0.617	0.469	7.47	0.118	0.025	604	114	0.0829			

CPG Hip2 parameters											
Name	V	y	Af	ε	Wc	Wi	$Wsiar$	$PF\alpha$	$PF\theta$		
Value	0	14.8	1.85	1.14	0.2	0.345	0.0806	7.53	0		
Name	Wgb	Wgf	$Wsh1$		$MN\alpha$	$MN\theta$	Wpf	Wpe		$Wsia$	$Wsib$
Value	0	0	0		21.4	7.03	18.1	39.4		11.7	2.61
Name	$Wsii$	Wm	$SNIa\alpha$	$SNIa\theta$	$SNIb\alpha$	$SNIb\theta$	$SNII\alpha$	$SNII\theta$			
Value	0.633	0.171	15.9	0.198	0.0816	325	52	0.104			

CPG Knee parameters											
Name	V	y	Af	ε		Wi	$Wsiar$	$PF\alpha$	$PF\theta$		
Value	0	8.16	0.684	0.693		0.1	0.172	16.6	0		
Name			$Wsh2kf$	$Wsh2ke$	$MN\alpha$	$MN\theta$	Wpf	Wpe		$Wsia$	$Wsib$
Value			0	0	18.9	4.6	27.5	22.1		11.7	6.16
Name	$Wsii$	Wm	$SNIa\alpha$	$SNIa\theta$	$SNIb\alpha$	$SNIb\theta$	$SNII\alpha$	$SNII\theta$			
Value	0.25	0.724	7.47	0.363	0.0847	293	30	0.241			

CPG Ankle parameters											
Name	V	y	Af	ε		Wi	$Wsiar$	$PF\alpha$	$PF\theta$	Wfb	Wff
Value	0	4	1.33	0.564		0.241	0.136	13.9	0	0	0
Name	Wgb	Wgf	$Wsh2a$		$MN\alpha$	$MN\theta$	Wpf	Wpe	$Weii$	$Wsia$	$Wsib$
Value	1.08	1.25	0		13.3	5.25	24.5	36.2	0	13.6	7.02
Name	$Wsii$	Wm	$SNIa\alpha$	$SNIa\theta$	$SNIb\alpha$	$SNIb\theta$	$SNII\alpha$	$SNII\theta$			
Value	0.51	0.296	15.9	0.259	0.0663	250	81.4	0.109			

Appendix B

OpenSim

"osim-rl" project (`http://osim-rl.stanford.edu`) has helped me greatly in initial understanding of OpenSim[15].

Simulation platform with OpenSim is implemented in Python utilizing SWIG wrapper (`http://www.swig.org`) to call OpenSim's C++ functions. Each time-tick the controller calculates its output from previous state and feedback signals. Each CPG provides excitation values for two corresponding antagonist groups of muscles that are passed to musculoskeletal model. Musculoskeletal model then integrates its state to the next time-tick, producing kinematic and dynamic values of simulated movement.

Unlike GAIT2DE, OpenSim contains its own integrator. To initialize the simulator and get the required state and feedback values from model the user must call numerous functions. All this was implemented as a Python class `msmg.py` (MusculoSkeletal Model of Gait).

Generally, one must initialize the model and assign a control function to each actuator, either muscle or torque actuator. Then, at each time-tick, one applies control values to actuators, then integrate to the target time. Optionally, to get contact forces, it is required to call `model.realizeAcceleration(state)`. And finally, get all the required feedback values.

The simulation can be rendered using SimTK visualizer, which unfortunately is not wrapped for Python (this is true for all SimTK functionality), so the actions can't be automated except visualizer's window open and close.

[15] I am grateful to my colleague Mélanie Jouaiti who has shown me this project

Table B.1: Parameters of 8-CPG gait controller with OpenSim simulator.

Common parameters							
Name	ρ_i	ρ_c	τ_m	τ_s	σ_f	Af	Wi
Value	0.1	0.1	0.35	3.5	2	1	0.1
Name	Wc	σ_s	ε	$SNF\alpha$	$SNF\theta$	$SNG\alpha$	$SNG\theta$
Value	0	20	2	600	0.025	10	0.5

CPG Hip1 parameters										
Name	V	y	$i_{inj}P$	$PF\alpha$	$PF\theta$	Wfb	Wff	Wgb	Wgf	
Value	1	3	0.975	25.75	0.8179	5.246	11.17	0.8187	2.834	
Name	$MN\alpha$	$MN\theta$	Wpf	Wpe	$Wsia$	Wm	$SNIa\alpha$	$SNIa\theta$		
Value	9.669	5.012	44.01	9.698	0.2337	1.187	3000	0.01		
CPG Hip2 parameters										
Name	V	y	$i_{inj}P$	$PF\alpha$	$PF\theta$			Wgb	Wgf	$Wsh1$
Value	1	-1	1.67	35.05	1.753			23.37	6.566	1.865
Name	$MN\alpha$	$MN\theta$	Wpf	Wpe	$Wsia$	Wm	$SNIa\alpha$	$SNIa\theta$		
Value	9.971	0.8913	51.49	1.813	0.04789	0.8374	3000	0.011		
CPG Knee parameters										
Name	V	y	$i_{inj}P$	$PF\alpha$	$PF\theta$				$Wsh2kf$	$Wsh2ke$
Value	0.5	0	1.69	11.15	0.9901				4.229	2.495
Name	$MN\alpha$	$MN\theta$	Wpf	Wpe	$Wsia$	Wm	$SNIa\alpha$	$SNIa\theta$		
Value	9.087	1.096	4.894	1.765	0.4414	0.2037	3000	0.011		
CPG Ankle parameters										
Name	V	y	$i_{inj}P$	$PF\alpha$	$PF\theta$	Wfb	Wff	Wgb	Wgf	
Value	-0.5	2	5.71	35.62	0.8348	1.121	2.006	2.108	3.23	
Name	$Wsh2a$	$MN\alpha$	$MN\theta$	Wpf	Wpe	$Weii$	$Wsia$	Wm	$SNIa\alpha$	$SNIa\theta$
Value	0.9171	8.935	2.977	6.175	7.428	0.1883	13.6	0.1347	3000	0.011

Table B.2: Parameters of 4-CPG gait controller with OpenSim simulator.

Muscle weights							
Name	addmagIsch	bflh	bfsh	edl	ehl	fdl	fhl
Value	0.05	0.22	0.4	0.43	0.16	0.024	0.074
Name	gaslat	gasmed	glmax1	glmax2	glmax3	grac	iliacus
Value	0.19	0.37	0.2	0.25	0.047	0.23	0.85
Name	perbrev	perlong	psoas	recfem	sart	semimem	semiten
Value	0.02	0.049	0.58	0.29	0.31	0.35	0.31
Name	soleus	tfl	tibant	tibpost	vasint	vaslat	vasmed
Value	0.46	0.45	0.79	0.075	0.064	0.15	0.096

Common CPG parameters							
Name	ρ_i	ρ_c	τ_m	τ_s	σ_f	Af	Wi
Value	0.1	0.1	0.35	3.5	2	1	0.1
Name	Wc	σ_s	ε	$SNF\alpha$	$SNF\theta$	$SNG\alpha$	$SNG\theta$
Value	0	20	2	+5.1184e+01	-2.5556e-02	0.03	0

CPG HK parameters							
Name	V	y	$i_{inj}P$	$PF\alpha$	$PF\theta$	Wfb	Wff
Value	1	0	0.975	13.085	0	+6.4512e-01	+3.8063e-01
Name	Wgb	Wgf					
Value	+2.0724e-01	+9.9489e-01					
Name	$MN\alpha$	$MN\theta$	Wpf	Wpe	$Wsia$	$SNIa\alpha$	$SNIa\theta$
Value	12	0	2	2	0.2	200	0.2
CPG KA parameters							
Name	V	y	$i_{inj}P$	$PF\alpha$	$PF\theta$	Wfb	Wff
Value	0	3	5.71	15	0	+4.7609e-01	+7.7329e-01
Name	Wgb	Wgf	$Wshkf$	$Wshke$			
Value	+8.9878e-01	+8.3047e-01	+4.9658e-01	+9.1930e-01			
Name	$MN\alpha$	$MN\theta$	Wpf	Wpe	$Wsia$	$SNIa\alpha$	$SNIa\theta$
Value	30	0	2	2	0.2	200	0.2

Appendix C

Résumé détaillé

Le travail présenté dans cette thèse concerne les domaines de la biomécanique, de la modélisation neuronale et du traitement du signal. Il vise à simuler l'impact des troubles moteurs sur la démarche humaine pour aider au diagnostic non invasif des maladies neurodégénératives telles que la maladie de Parkinson. L'idée principale est d'évaluer la coordination des mouvements du patient pour prédire son état après un acte chirurgical en fournissant des données biologiques, biomécaniques et cinématiques qui seraient difficiles à collecter autrement.

Plus particulièrement, le but de cette thèse est de créer une nouvelle plate-forme de simulation neuro-musculo-squelettique du système locomoteur humain pour reproduire des allures de marche saines ou altérées par la maladie de Parkinson ou par des troubles du système musculo-squelettique ou des troubles locomoteurs.

La thèse est organisée en cinq chapitres:

[1] Le premier chapitre introduit la démarche saine et déficiente, et décrit le contrôle moteur humain, les causes de la maladie de Parkinson et les modèles neuronaux impliqués.

[2] Le second concerne les central pattern générations (CPG) dédiés à la modélisation des structures neuronales spinales qui contrôlent les mouvements rythmiques.

[3] Le troisième chapitre présente un modèle musculo-squelettique du corps humain implémentés dans deux simulateurs dynamiques utilisés en général pour évaluer les commandes motrices et modifié par nos soins pour fournir en retour des informations proprioceptives et extéroceptives.

[4] Le quatrième concerne l'analyse des marches reproduites par notre simulateur et l'optimisation des paramètres du contrôleur de marche proposé.

[5] Le cinquième présente les résultats des simulations neuro-musculo-squelettiques, produites par notre architecture de contrôle à base de CPGs et les différentes allures de marche produites.

Enfin, la thèse se termine par une conclusion qui résume le travail, discute des résultats et présente les futures orientations de travail dans des perspectives à long terme et à court terme.

C.1 Marches saines et déficientes

Le premier chapitre passe en revue les grands principes du système nerveux qui contrôle la locomotion humaine en se concentrant sur les structures neuronales situées dans le cerveau et qui sont à l'origine des troubles parkinsoniens. Afin de développer un contrôleur neuronal capable d'activer des muscles simulés en boucle fermée pour générer des allures de marche, ce chapitre examine les signaux de contrôle de sortie produits par des modèles connus de structures cérébrales pertinentes pour la locomotion.

Premièrement, il aborde brièvement le problème de la classification cinématique de la marche pour différentes maladies, y compris neurodégénératives, qui affectent la locomotion au niveau du cerveau et non au niveau de la colonne vertébrale ou en dessous. Ensuite, le chapitre décrit comment les signaux de contrôle sont transmis dans la moelle épinière pour contrôler l'activité musculaire à travers plusieurs boucles fermées hypothétiques, car ces boucles n'étant pas bien définies dans la littérature.

La description de la voie des signaux de locomotion commence à partir des noyaux gris centraux du cortex ou ganglions de la base (BG), qui sélectionnent le comportement locomoteur et intègre les données sensorielles. Ensuite, plusieurs autres régions du cerveau [15] transmettent des signaux de contrôle aux structures de la moelle épinière — les central pattern generators (CPG). Les relations entre les noyaux GB sont également hypothétiques, mais dans cette thèse, nous proposons un schéma de connexion GB commun, compilé à partir de plusieurs travaux, et ayant une description plus détaillée des chemins des GB. Un noyau GB particulièrement important pour ce travail est la substance noire « pars compacta » (SNc), qui fournit de la dopamine (DA) aux GB. La DA contribue au contrôle de la motricité fine en régulant l'activité des autres noyaux des GB. La maladie de Parkinson (MP) est caractérisée par la mort des neurones dopaminergiques dans le SNc. L'épuisement de la dopamine produit une hypokinésie (diminution des mouvements corporels), une réduction globale de la puissance motrice. Inversement, les médicaments utilisés pour traiter la maladie de Parkinson peuvent produire une activité dopaminergique excessive, ce qui permet d'activer les systèmes moteurs à des moments inappropriés et donc de provoquer une dyskinésie (mouvements musculaires involontaires).

La maladie de Parkinson a été choisie pour nos simulations pour son effet notable sur la démarche : tremblements, rigidité musculaire, mouvements lent et imprécis et le symptôme appelé « Freezing of Gait » (FoG). Le FoG se caractérise par de courts épisodes d'incapacité à faire un pas ou des pas très courts lors de l'initiation de la marche ou du retournement [15]. Ce symptôme est facile à étudier et à compenser partiellement en appliquant divers signaux aux patients [29, 30, 31, 32, 33]. Les modèles des circuits neuronaux de type GB et des structures associées poursuivent généralement la modélisation d'une tâche de prise décision ou sont issus de données de patients. Ces derniers modèles, sont très difficiles à exploiter pour contrôler la marche. Ils entraînent des paramètres intrinsèques du système, tels que les taux de décharge des neurones, qui ne correspondent pas directement aux commandes de mouvement. D'autres modèles proposés simulent l'écriture manuscrite [19], le levage d'objets [35] et la marche [36].

Ce travail de thèse considère six modèles GB de fonction de prise de décision avec simulation de la MP appliquée via un paramètre de niveau DA spécial ou en ajustant les paramètres aux données expérimentales de patients parkinsoniens. Les trois premiers modèles simulent des oscillations anormales dans les GB, liées à la MP pour l'étudier et prédire les conséquences des changements de structure de connexion dans les GB. Le résultat de ces modèles est une activité neuronale qui doit être transformée de manière non triviale en commandes de contrôle pour le réseau moteur (motoneurones). Les trois

modèles suivants contiennent des modules de prise de décision, similaires aux GB, pour effectuer des tâches d'écriture manuscrite, de prise d'objet et de marche, tout en simulant l'effet de la MP. Le modèle d'écriture manuscrite produit des positions cibles relatives des pointes de stylo sur trois axes à partir du modèle d'une partie du cortex, des noyaux gris centraux et des circuits moteurs. La simulation de la MP est réalisée sous forme de déplétion de DA via un paramètre des neurones de type Hodgkin-Huxley. Le travail contient une boucle de contrôle moteur complète, dont les parties supérieures sont utilisables comme contrôleur musculaire. Le modèle de prise d'objet permet de rendre compte avec précision des résultats de patients normaux témoins et de parkinsoniens à partir de deux approches : avec et sans médication, et les sujets témoins soulevant des blocs de 300 g, certains recouverts de soie ou de papier de verre. Le dernier travail simule des modèles de vitesse de marche altérés par la MP et basés sur 2 études sur le FoG, patients avec et sans médicaments dopaminergiques franchissant une porte avec une largeur variable. Ce travail utilise un modèle CPG qui imite le rythme rachidien pour calculer la distance parcourue.

Les modèles de contrôle moteur supérieur considérant des structures cérébrales telles que la glycémie, pourraient à l'avenir servir de contrôleur de prise de décision pour la marche altérée ainsi que de contrôle du comportement d'une marche saine. Bien qu'elle n'endommage pas directement le système locomoteur, la MP affecte les signaux de commande, ce qui entraîne des troubles de la marche. La simulation fonctionnelle de ces structures décisionnelles a été jugée suffisante pour le contrôleur spinal de la marche, discutée dans les chapitres suivants.

C.2 Central pattern generators

Ce deuxième chapitre présente le contrôle neuronal développé pour la plateforme de simulation de marche. Le contrôleur est basé sur un modèle original de central pattern generators (CPG) inspiré du réseau locomoteur spinal. Les GPC biologiques sont des réseaux d'inter- et de motoneurones situés dans la moelle épinière qui contrôlent les activités rythmiques telles que la respiration, la mastication, la déglutition, la démarche, etc. [39, 40, 41, 42, 43]. Un CPG génère de manière autonome une activité rythmique, même après avoir été déconnecté du contrôleur supérieur et des afférences sensorielles. Les CPG reçoivent des signaux issus du haut niveau par des voies extrapyramidales dans le tronc cérébral [14]. Dans ce travail, ce contrôleur de haut niveau est appelé MLR (région locomotrice mésencéphale). On pense que cette région du cerveau initie la locomotion dans la chaîne de prise de décision concernant les mouvements autonomes [43, 55]. En tant que rétroaction sensorielle des muscles et de l'environnement, les signaux biologiquement plausibles sont utilisés pour contrôler la démarche humaine au niveau de la colonne vertébrale. Seul le signal vestibulaire des signaux récepteurs utilisés provient du cerveau. Les autres signaux sont des capteurs proprioceptifs musculaires et cutanés des types *Ia*, *Ib* et *II* [14].

Le modèle biologique de CPG n'est pas encore établi et fait l'objet d'une recherche active [42, 58, 44, 59]. Les chercheurs ont proposé un certain nombre de modèles à différentes échelles pour reproduire les caractéristiques spécifiques connues des CPG. Ce travail passe en revue plusieurs modèles de CPG.

Les modèles microscopiques simulent le comportement CPG au niveau cellulaire. Ils sont utilisés pour l'étude des fonctions et des régimes de groupes spécifiques de neurones dans différents segments de la moelle épinière [14, 39]. Ces modèles utilisent différentes variantes du modèle de neurones Hodgkin-

Huxley pour les circuits oscillants. Ces modèles microscopiques de CPG sont utilisés pour modéliser des réseaux neuronaux ressemblant à des CPG naturels [41, 63, 64] et pour étudier la neurophysiologie des CPG biologiques [65].

Les modèles mésoscopiques simulent le comportement fonctionnel des réseaux de CPG. Ils sont composés d'un ou plusieurs neurones d'architecture unique appelés générateurs de rythme (RG) et de certains périphériques ou interneurones autour de RG. Une caractéristique distinctive des modèles mésoscopiques est un réseau interconnecté de deux ou plusieurs RG qui produisent des oscillations. Parmi les principaux modèles mésoscopiques, il y a le CPG de Matsuoka [66], le CPG de Rowat et Selverston [40] et l'oscillateur de phase qui ont été utilisés par Ijspeert et al. [76].

Les modèles macroscopiques de CPG offrent une seule unité entière produisant les motifs rythmiques souhaités. L'oscillateur de Hopf, tres répandu, a été utilisé par Righetti & Ijspeert [75] comme CPG pour contrôler un robot bipède. Un modèle de CPG construit comme une machine à état, a aussi été appliqué par Jo et al. [77] pour contrôler cinq "phases" du cycle de marche.

Tous les modèles de CPG mentionnés ont été appliqués dans divers domaines. Malgré l'origine du CPG élément de la structure de différents groupes de neurones dans les segments de la moelle épinière, les modèles de CPG artificiel sont également largement utilisés dans le contrôle bio-inspiré des robots. Bien que l'objectif soit biologique, pour contrôler les muscles, une approche plus fonctionnelle a été recherchée. À l'échelle mésoscopique, le CPG n'a pas besoin d'être structurellement biologique, mais uniquement bio-inspiré. Dans ce travail de thèse, une version modifiée du modèle de CPG de Rowat-Selverston a été utilisée pour générer les excitations musculaires du modèle musculo-squelettique humain. Le modèle de CPG utilisé est soutenu par deux études neurophysiologiques [41, 87] et combine les différents neurones dans une circuiterie de CPG multicouches multi-modèles.

L'architecture est basée sur les travaux de Rybak et al. [41] pour un CPG à deux niveaux qui sépare le timing et l'activation du cycle de locomotion. La figure C.1 présente le schéma général du CPG à deux couches, plus les motoneurones et les retours sensoriels afférents avec une architecture de type demi-centre. Les oscillations sont générées au niveau de la couche supérieure dite « Rhythm generation » (RG) puis transmises à la couche « pattern formation » (PF), qui projette sur les motoneurones (MN). De plus, le modèle inclut explicitement le contrôleur supérieur et les neurones sensoriels de rétroaction (SN) qui façonnent l'activité des neurones CPG. Ces neurones sont de deux types: les propriocepteurs de la boucle de rétroaction musculaire (type *Ia*, *Ib* et *II*) et les extérocepteurs de l'environnement (angle du corps, force pied/sol).

Les modes et l'activité du modèle sont contrôlés par un contrôleur de niveau supérieur situé dans le cerveau (par exemple MLR) qui fait varier la fréquence des motifs générés, qui réinitialise la phase et qui peut bloquer le signal de contrôle.

Le modèle de neurone oscillant de Rowat-Selverston peut être réécrit comme un oscillateur de Van der Pol généralisé afin d'y insérer une règle d'apprentissage hebbien [72]. Deux neurones couplés inhibiteurs forment alors une unité oscillante (couche RG) du CPG.

La fonction d'activation des neurones est sigmoïdale, tels que proposé auparavant par Nassour et al. [82]. Les neurones sensoriels (SN), à la fois proprioceptifs et extéroceptifs, fournissent une rétroaction au contrôleur. Les neurones proprioceptifs renvoient les informations de l'intérieur du corps, tels que les valeurs articulaires ou musculaires, tandis que les neurones extéroceptifs réagissent aux changements

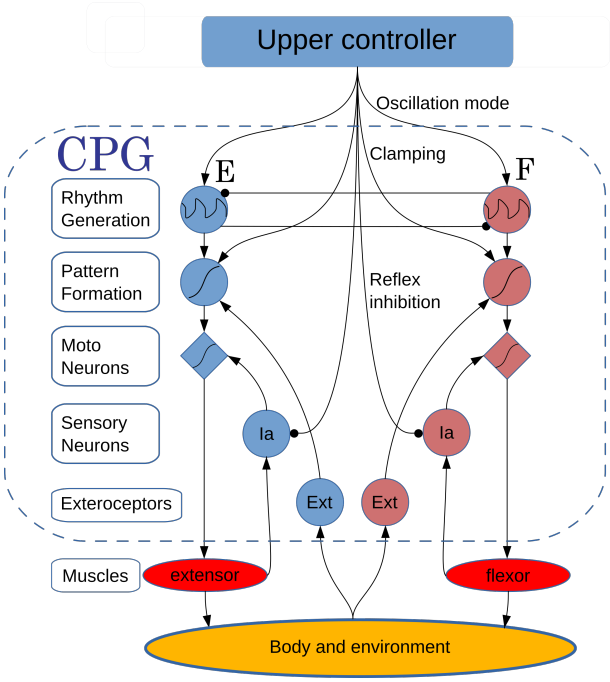


Figure C.1: Le modèle CPG avec cinq couches et un contrôle supérieur.

environnementaux tels que l'angle corporel ou la force de réaction au sol.

Les quatre couches (Rhythm Generator, Pattern Formation, Motoneuron et Sensory Neuron) forment une unité CPG, qui est influencée par une unité de contrôle supérieure. L'unité de commande supérieure déclenche/arrête la locomotion et change le mode, par ex. marcher courir. Plusieurs CPG interconnectés forment un réseau qui contrôle les muscles dans un simulateur musculo-squelettique. Pour fermer la boucle de contrôle, le simulateur fournit des signaux de rétroaction, mesurés par les neurones sensoriels. À la fin du chapitre, deux contrôleurs supplémentaires sont décrits, visant à aider à maintenir l'équilibre pendant la marche : un contrôleur réflexe et un contrôleur proportionnel-dérivé des bras. Le premier contrôleur modélise des arcs réflexes monosynaptiques des muscles des jambes en utilisant uniquement des capteurs proprioceptifs et extéroceptifs. Le second contrôleur utilise le principe proportionnel-dérivé pour contrôler les bras du modèle c'est-à-dire l'articulation de l'épaule en fonction du mouvement des hanches.

C.3 Modèle musculo-squelettique du corps humain

Ce troisième chapitre concerne les simulateurs musculo-squelettiques utilisés dans cette thèse. Il décrit également les modifications apportées aux simulateurs pour obtenir une simulation physique en boucle fermée du système locomoteur de la marche humaine sur le sol. La rétroaction sensorielle proprioceptive et extéroceptive des simulateurs est utilisée par les CPG. Dans un premier temps, le simulateur musculo-squelettique GAIT2DE a été utilisé pour sa simplicité, puis la méthode et les modèles ont été implémentés dans le simulateur OpenSim qui est plus réaliste et plus utilisé dans le domaine de la biomécanique.

Le simulateur de système locomoteur humain «GAIT2DE» simule l'activité musculaire et son action

sur le squelette pour actionner les mouvements des jambes dans le plan sagittal, en tenant compte des efforts dynamiques internes et externes. Le modèle locomoteur humain comprend 7 segments corporels et 16 muscles (Fig. C.2, à gauche). Il est implémenté en tant que fonction MATLAB MEX. Le modèle tient compte des phénomènes physiques (frottement au sol, forces externes, dynamique des membres, etc.).

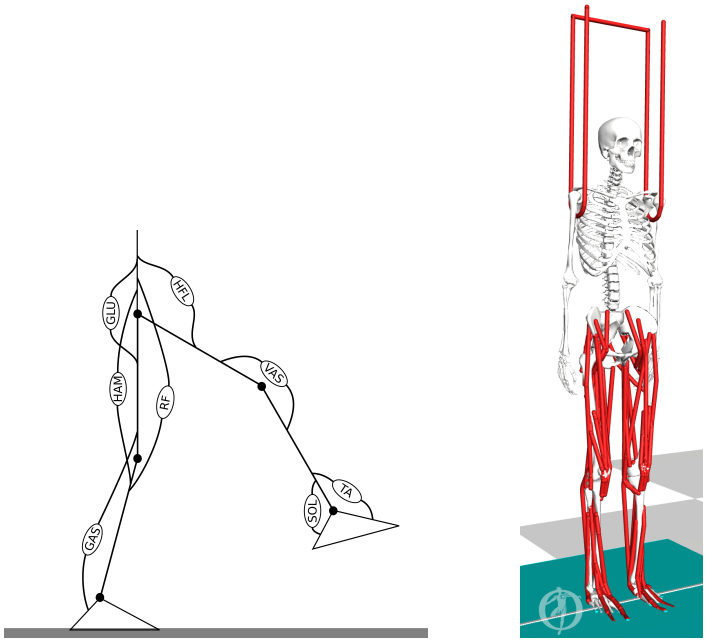


Figure C.2: À gauche: modèle GAIT2DE avec muscles d'une jambe. À droite: modèle OpenSim modifié.

Pour poursuivre nos travaux, le simulateur OpenSim [5] développé par l'université de Stanford a été utilisé. OpenSim est un outil disponible gratuitement pour la modélisation et la simulation de mouvement. Il est utilisé en réadaptation, orthopédie, robotique, ergonomie, l'analyse de performance et la conception de systèmes portables. OpenSim a été utilisé pour la mise en œuvre du modèle musculo-squelettique de l'homme par Rajagopal et al. [4] (Fig. C.2, à droite). Ce modèle possède 37 degrés de liberté, 22 éléments corporels, 80 muscles (modèle de Hill) qui contrôlent 8 articulations des jambes et 17 contrôleurs de couple qui contrôlent les 7 articulations du reste du corps. Les auteurs l'ont développé ce modèle sur la base de mesures anatomiques obtenues sur 21 cadavres et à partir d'images de résonance magnétique de 24 jeunes sujets sains. Le codage Python a été utilisé sur OpenSim.

Le modèle musculo-squelettique original a été conçu pour les calculs de dynamique inverse, il manque donc quelques fonctionnalités nécessaires à nos travaux que nous avons dû développer, particulièrement la mesure du contact avec le sol, la limitation des angles articulaires et quelques simplifications de la structure du modèle.

C.4 Analyse de marche simulée

Le chapitre quatre s'intéresse à l'analyse de la marche simulée et à l'optimisation des paramètres du contrôleur en comparant les résultats de la simulation à la marche réelle. L'analyse consiste en une description du cycle de marche afin de décomposer les données de marches humaines simulées et réelles.

Concernant la symétrie des mouvements des jambes droite et gauche, on peut diviser le cycle de marche en plusieurs phases en se basant sur des événements cinématiques des jambes. Dans ce travail, le cycle de marche habituel a été étendu à 10 événements de marche symétriques afin d'identifier les caractéristiques de marche simulées et mesurées à partir de la période globale de la marche et de la phase initiale, des caractéristiques de chaque phase de la marche d'une variation quantifiable des cycles de marche.

Pour la comparaison et l'évaluation des modèles, les valeurs cinématiques des membres et l'activation des muscles pouvant être mesurés par électromyographie (EMG) ont été utilisées. Pour la procédure de comparaison avec un prototype de modèle GAIT2DE, des courbes moyennes des angles articulaires et l'activation musculaire durant un cycle de marche d'une femme normale de 22 ans ont été utilisées. Pour une utilisation avec le modèle OpenSim, les valeurs neuromusculaires des auteurs du modèle ont été utilisées. Ils avaient été obtenus à partir d'un calcul dynamique inverse d'un sujet de sexe masculin âgé de 31 ans effectuant un cycle de marche unique à une vitesse de confort [4].

L'optimisation prend en compte les valeurs des paramètres CPG et les poids des connexions tandis que le modèle musculo-squelettique est considéré comme inchangé. Le contrôleur de marche contient de 44 à environ 130 paramètres. Pour l'optimisation, le module BlackBox de Python pour l'optimisation parallèle de fonctions de type boîte noire, a été utilisé [106]. BlackBox reconstruit une surface multidimensionnelle à partir de la fonction cible avec un nombre limité d'appels. Ensuite, il recherche le minimum avec un autre nombre limité d'appels à la fonction.

En tant que fonction cible, une approche directe a été utilisée: comparer les paramètres cinématiques et cinétiques de la marche simulée avec des données mesurées sur l'homme en utilisant la corrélation croisée. La procédure de comparaison a été mise en œuvre à des fins d'optimisation du contrôleur et d'une classification très simple des allures de marche. La fonction cible d'optimisation évalue la qualité globale de l'allure de marche en comparant les valeurs dynamiques de simulation aux valeurs humaines correspondantes pendant la marche. Le classiffieur décrit la simulation de la marche avec des valeurs de comparaison et avec en plus des valeurs de période de marche et de phase initiale.

C.5 Résultats des simulations neuro-musculo-squelettiques

Le chapitre cinq présente les résultats obtenus avec les simulateurs OpenSim et GAIT2DE en intégrant une circuiterie complète à base de CPG et un contrôle réflexe d'équilibre basé sur une boucle de rétroaction proprioceptive. Le chapitre se compose de cinq sections, chacune décrivant un contrôleur. Tout d'abord, un CPG simple dédié au contrôle d'une articulation est présenté afin de mieux comprendre son architecture, les paramètres. Les capacités de simulation de transfert (extension/flexion) de la jambe en utilisant 2 muscles avec 1 CPG à vitesse constante et changeante est montrée. Le modèle utilise l'angle de l'articulation de la hanche comme rétroaction pour limiter en douceur l'amplitude du transfert. Cet exemple montre, en simulation, la capacité du CPG à générer des mouvements rythmiques pour contrôler les muscles d'une articulation biologique et à varier ses paramètres selon des signaux qui peuvent provenir du niveau supérieur. Les sections suivantes montrent les circuiteries de plusieurs CPG pour contrôler les membres inférieurs du corps humain et simuler la marche.

Les trois contrôleurs CPG permettent des simulations de marche, dont l'architecture et les rôle varient. Ils consistent en:

[1] Le contrôleur d'articulation. Il met en œuvre un contrôle de type robotique avec des muscles monoarticulaires à l'aide de 6 CPG. Les résultats montrent des simulations de marche à vitesse constante et variable simulées avec le modèle GAIT2DE. Cet solution utilise une rétroaction biologiquement plausible des capteurs de type Ia du muscle, de la force de réaction au sol et de l'angle du tronc. Chaque CPG contrôle sa propre paire de muscles et a son propre rôle en fonction rôle de l'articulation. Les profils des angles articulaires obtenus sont qualitativement similaires à ceux du cycle de marche humaine [104, 110].

[2] Le contrôleur de muscles. C'est le contrôleur le plus flexible, favorisant un CPG pour chaque groupe de muscles antagonistes, y compris bi-articulaires. En utilisant les modèles GAIT2DE et OpenSim, les muscles sagittaux ont été divisés en 16 groupes, contrôlés par 8 CPGs. Cela permet de reproduire un exemple de marche, interrompue par une commande externe au CPG, modélisant un épisode de FoG Parkinsonien. Ce contrôleur utilise tous les signaux de rétroaction musculaire de type *Ia*, *Ib* et *II*. Le contrôleur obtenu, avec l'ensemble des paramètres optimisés, entraîne une démarche stable lente avec une assez grande foulée. Différentes valeurs de paramètres permettent de modifier le type de résultat, la vitesse de marche et la longueur de foulée. Les paramètres de chaque CPG peuvent être modifiés indépendamment pour simuler différents types d'allure de marche, allant de pas lents et étroits observés chez les patients parkinsoniens à une allure de jogging à sauts larges.

[3] Le contrôleur de phase. Ce contrôleur vise à réduire le nombre de CPG et donc le nombre total de paramètres en faisant en sorte que chaque CPG contrôle une phase d'activation musculaire. Cette approche réduit la flexibilité du contrôle mais cela peut être amélioré en appliquant une approche du contrôle basée sur le principe de synergie musculaire. Cette question est examinée dans le chapitre conclusion. La simulation montre la phase d'initiation de l'allure de marche depuis la phase debout en double support en inhibant le contrôleur CPG puis en libérant le contrôleur d'équilibre et en poursuivant la marche.

Le chapitre se termine par une démonstration du contrôleur réflexe utilisé pour maintenir l'équilibre au début et à la fin de la marche. Ce contrôleur utilise les mêmes signaux de rétroaction biologique que le contrôleur CPG et est inhibé pendant la marche par un signal particulier supposé venir du cerveau. La mise en œuvre du contrôleur permet de contrôler l'équilibre pendant environ 3 secondes suivies du temps correspondant à l'extension des chevilles pendant lequel le corps est debout sur les orteils tout en tombant vers l'avant, ce qui est un réflexe final de maintient d'équilibre avant de faire le pas réflexe en avant. Ces résultats montrent qu'il est possible de générer différents patrons de marche qui sont relativement stables et coordonnés en modifiant les paramètres neuronaux des GPC, reproduisant ainsi les schémas observés pour la maladie de Parkinson ou d'autres troubles de la marche bien connus en médecine. Cette la plate-forme peut également simuler divers rythmes dus à la dégénérescence musculaire.

C.6 Conclusions générales

Le dernier chapitre concerne les conclusions, la discussion des résultats et les perspectives. Il résume le travail développé dans la thèse et propose quelques améliorations dans les circuits CPG et des modifications au modèle musculo-squelettique dans OpenSim. L'accent est mis sur la plausibilité biologique des

modèles utilisés ainsi que sur les résultats obtenus des allures de marche simulées.

La seconde partie du chapitre est une section de discussion avec une sous-section spéciale sur les perspectives de synergie musculaire. Les perspectives générales concernent la plate-forme de simulation, c'est-à-dire un ensemble qui permettra de simuler une démarche anormale due à différentes causes telles que les maladies neurodégénératives ou l'impact de l'ajout de membres artificiels (prothèses) et d'interventions chirurgicales. L'hypothèse de synergie musculaire est une manière de contrôler le système moteur, visant à résoudre le problème de la redondance du contrôle musculaire. Il est mis en évidence en raison de ses perspectives dans le contrôle à base de CPG, particulièrement dans le contrôle des phases, où la synergie est censée étendre les limites des mouvements de quelques CPG. Les CPG sont alors proposés comme des générateurs de salves de primitives motrices, obtenues à partir des enregistrements EMG musculaires.

Le chapitre se termine par la section « travaux futurs », qui traite des orientations de développement possibles pour la plate-forme de simulation. Tout d'abord, un meilleur contrôle de l'équilibre est nécessaire pour une marche lente. Ensuite, le contrôleur supérieur doit être mieux étudié à la fois pour un contrôle sain de la marche et pour reproduire plus fidèlement les allures affectées par des maladies neurodégénératives.

La section se termine par quelques perspectives à court terme sont présentées, elles sont simples et assez rapide à développer et à implémenter. Enfin, les perspectives à long terme sont présentées, proposant une orientation future du travail, comme la recherche de différents types d'allures de marche ou l'extension des modèles et de l'architecture du contrôleur.

Résumé

Mon travail de thèse vise à simuler l'impact des troubles moteurs sur la démarche de l'homme afin d'aider au diagnostic non invasif des maladies neurodégénératives telles que la maladie de Parkinson. En effet, la simulation du système locomoteur humain contribue à approfondir notre compréhension du fonctionnement du corps humain en fournissant des données biologiques, biomécaniques et cinématiques qu'il serait difficile de recueillir autrement et en aidant à évaluer la coordination des mouvements d'un patient et à prédire son état après une intervention chirurgicale.

Le but de ma thèse est plus particulièrement de créer une nouvelle plate-forme de simulation neuro-musculo-squelettique du système locomoteur humain permettant de reproduire des allures de marche saines ou altérées par la maladie de Parkinson, ou par des troubles du système musculosquelettique ou des troubles locomoteurs

Le travail présenté comprend six chapitres.

Le premier chapitre passe en revue les grands principes du système nerveux qui contrôle la locomotion humaine en se focalisant sur les circuits nerveux qui impliquent des structures neuronales situées dans le cerveau et où sont les sources des troubles parkinsoniens. Ensuite, on décrit comment les signaux de contrôle sont transmis à la moelle épinière pour contrôler l'activité musculaire au travers de plusieurs boucles fermées. Différents modèles neuronaux des troubles parkinsoniens issus de la littérature et en lien avec nos travaux sont présentés dans ce chapitre.

Le second chapitre présente le contrôle neuronal de la future plate-forme de simulation. Ce contrôleur est basé sur un modèle original de « central pattern generators » (CPG) inspiré du réseau locomoteur spinal et développé au LORIA depuis quelques années. Ce modèle de CPG peut générer des signaux rythmiques variables selon ses paramètres neuronaux intrinsèques, lesquels sont contrôlés par des signaux descendants provenant du module de prise de décision modélisant le comportement des noyaux gris centraux. Les signaux de sortie des motoneurones du CPG sont appliqués en tant qu'excitation au modèle de muscles flexeur/extenseur.

Le troisième chapitre concerne les simulateurs musculosquelettiques utilisés dans cette thèse ainsi que les modifications apportées pour obtenir une simulation physique en boucle fermée du système locomoteur marchant sur le sol et dont les retours sensoriels proprioceptifs et extéroceptifs sont exploités par les CPGs. Dans un premier temps, le simulateur musculosquelettiques GAIT2DE a été utilisé pour des raisons de simplicité, puis la méthode et les modèles ont été implémentés dans le simulateur OpenSim qui est plus réaliste et plus utilisé en Biomécanique.

Le chapitre quatre est consacré à l'analyse de marche simulée et l'optimisation des paramètres du contrôleur. La partie analyse présente le cycle de marche humaine et une méthodologie pour extraire les données de marche simulées et réelles. Le cycle de marche est utilisé de deux façons. D'une part, pour comparer des données de simulation avec des paramètres de marche réelle. D'autre part, dans la méthode d'optimisation du contrôleur basée sur l'analyse comparative utilisant la corrélation croisée.

Le chapitre cinq présente les résultats obtenus avec les simulateurs OpenSim et GAIT2DE en inté-

grant une circuiterie complète à base des CPGs, et en insèrent un réflexe du contrôle d'équilibre vers l'avant ou vers l'arrière à partir de retours proprioceptifs. Ces résultats montrent qu'il est possible de générer différentes allures de marche plus ou moins stables ou plus ou moins coordonnées en modifiant les paramètres neuronaux des CPGs reproduisant ainsi les allures observées pour la maladie de Parkinson ou d'autres troubles de la marche bien connus en médecine. On montre finalement que cette plate-forme permet de simuler diverses allures dues à la maladie de Parkinson mais aussi due à la dégénérescence musculaire.

Le dernier chapitre concerne la conclusion et les perspectives. On y résume les travaux développés dans la thèse en proposant certaines améliorations dans la circuiterie CPGs et dans les modifications qu'il faudrait apporter à OpenSim. Les perspectives concernent la plate-forme de simulation dans son ensemble qui permettra de simuler des allures anormales dues à différentes causes telles que d'autres maladies neurodégénératives, mais aussi l'impact de l'ajout de membres artificiels (prothèses) et des interventions chirurgicales.

Mots-clés: Modelisation, Central Pattern Generator, Traitement du signal.

Abstract

My thesis aims to simulate the impact of motor disorders on the human gait to help non-invasive diagnosis of neurodegenerative diseases such as Parkinson's disease. Indeed, the simulation of the human locomotor system helps to deepen our understanding of the functioning of the human body by providing biological, biomechanical and kinematic data that would be difficult to collect otherwise and by helping to evaluate the coordination of a patient's movements to predict its condition after surgery.

The goal of my thesis is, more specifically, to create a new platform for neuro-musculoskeletal simulation of the human locomotor system to reproduce healthy or altered walking gaits by Parkinson's disease or by disorders of the musculoskeletal system or locomotor disorders.

The work presented includes six chapters.

The first chapter reviews the main principles of the nervous system that control human locomotion by focusing on neural structures located in the brain and which are the sources of parkinsonian disorders. Next, I describe how control signals are transmitted in the spinal cord to control muscle activity through several closed loops. Different neuronal models of parkinsonian disorders from the literature and related to our work are presented in this chapter.

The second chapter presents the neural control of the future simulation platform. This controller is based on an original model of central pattern generator (CPG) inspired by the spinal locomotor network and developed at LORIA in recent years. This CPG model can generate variable rhythmic signals according to its intrinsic neural parameters, which are controlled by downlink signals from the decision-making module modeling the behavior of the basal ganglia. CPG motoneuron output signals are applied as an excitation to the flexor / extensor muscles in the model.

The third chapter concerns the musculoskeletal simulators used in this thesis as well as the modifications made to obtain a closed-loop physical simulation of the locomotor system walking on the ground and

whose proprioceptive and exteroceptive sensory feedback is used by the CPGs. As a first step, the musculoskeletal simulator GAIT2DE was used for simplicity, then the method and models were implemented in the OpenSim simulator which is more realistic and more used in Biomechanics field.

Chapter four considers simulated gait analysis and controller parameter optimization. The gait analysis part consists of gait cycle explanation and its application for decomposing of simulated and real human gait data. The gait cycle is used in two ways. First, in comparison of simulation data with parameters of real gait. Second, as part of controller optimization method based on comparative analysis using cross-correlation.

Chapter five presents the results obtained with the OpenSim and GAIT2DE simulators by integrating a complete circuitry based on CPGs and a reflex controller of equilibrium based on proprioceptive feedback. These results show that it is possible to generate different walking patterns that are relatively stable and coordinated by modifying the neuronal parameters of GPCs, thus reproducing the patterns observed for Parkinson's disease or other well-known gait disorders in medicine. Finally, I show that this platform can simulate various paces due to Parkinson's disease or muscle degeneration.

The last chapter concerns the conclusion and the perspectives. It summarizes the work developed in the thesis by proposing some improvements in CPG circuitry and in the modifications that should be made to OpenSim. The perspectives concern the simulation platform as a whole that will allow to simulate abnormal gait due to different causes such as neurodegenerative diseases or the impact of the addition of artificial limbs (prostheses) and surgical interventions.

Keywords: Modeling, Central Pattern Generator, Signal analysis.